Song Recommendation System

1st Jagruti Chaudhari
Department of Information Technology
SSBT's COET, Bambhori
Jalgaon, India
jagrutichaudhari929@gmail.com

2nd Roshani Mulmule
Department of Information Technology
SSBT's COET, Bambhori
Jalgaon, India
Roshanimulmule2001@gmail.com

3rd Vrushali Baisane
Department of Information Technology
SSBT's COET, Bambhori
Jalgaon, India
vrushalibaisaneail@gmail.com

4th Vaishnavi Depura
Department of Information Technology
SSBT's COET, Bambhori
Jalgaon, India
depuravaishnavi@gmail.com

ABSTRACT

In our project, we will be using a sample data set of songs to find correlations between users and songs so that a new song will be recommended to them based on their previous history. We will implement this project using libraries like NumPy, Pandas. We will also be using Cosine similarity along with Count Vectorizer. Along with this, a front end with flask that will show us the recommended songs when a specific song is processed. Along with the rapid expansion of digital music formats, managing and searching for songs has become significant. Though music information retrieval (MIR) techniques have been made successfully in last ten years, the development of song recommendation systems is still at a very early stage. Therefore, this paper surveys a general framework and state-of-art approaches in recommending song. Two popular algorithms: collaborative filtering (CF) and content-based model (CBM), have been found to perform well. Due to the relatively poor experience in finding songs in long tail and the powerful emotional meanings in songs, two user-centric approaches: context-based model and emotion-based model, have been paid increasing attention. In this paper, three key components in song recommender user modelling, item profiling, and match algorithms are discussed. Six recommendation models and four potential issues towards user experience, are explained. However, subjective song recommendation system has not been fully investigated. To this end, we propose a motivation-based model using the empirical studies of human behaviour, sports education, music psychology. A song recommendation system was developed that can learn users preferences. The system can classify a wide range of stored music using automatic music content analyses. Users can opt for music according to their mood, using such words as "bright", "exciting", "quiet", "sad" and "healing".

- **INTRODUCTION**

With the explosion of network in the past decades, internet has become the major source of retrieving multimedia information such as video, books, and music etc.People has considered that music is an important aspect of their lives and they listen to songs, an activity they engaged in frequently. Previous research has also indicated that participants listened to song more often than any of the other activities (i.e. watching television, reading books, and watching movies). Song, as a powerful communication and self-expression approach, therefore,has appealed a wealth of research.

However, the problem now is to organise and manage the million of song titles produced by society. MIR techniques have been developed to solve problems such as genre classification, artist identification and instrument recognition. Since 2005, an annual evaluation event called Music Information Retrieval Evaluation exchange (MIRE) is held to facilitate the development of MIR algorithms. Additionally, song recommender is to help users filter and discover songs according to their tastes. A good song recommender system should be able to automatically detect preferences and generate playlists accordingly. Meanwhile,the development of recommender systems provides a great opportunity for industry to aggregate the users who are interested in song. More importantly,it raises challenges for us to better understand and model users preferences in song. Currently, based on users listening behaviour and historical ratings, collaborative filtering algorithm has been found to perform well. Combined with the use of content-based model, the user can get a list of similar songs by low level acoustic features such as rhythm, pitch or high-level features like genre,instrument etc. Some music discovery websites such as Last.fm2, Allmusic3, Pandora4 and Shazam5 have successfully used these two approaches into reality. At the mean-time, these websites provide an unique platform to retrieve rich and useful information for user studies. song is subjective and universal. It not only can convey emotion, but also can it modulate a listener's mood. The tastes in song are varied from person to person, therefore, the previous approaches cannot always meet the users needs. An emotion-based model and a context-based model have been proposed. The former one SSBT's College of Engineering and Technology, Bambhori, Jalgaon (MS) 2 recommends song based on mood which allows the user to locate their expected perceived emotion on a 2D valence-arousal interface.The latter one collects other contextual information such as comments, song review, or social tags to generate the playlist. Though hybrid song recommendation systems would outperform the conventional models, the development is still at very early stage. Due to recent studies in psychology, signal processing,machine learning and musicology, there is much room for future extension. This paper, therefore, surveys a general song recommender framework from user profiling, item modelling, and item-user profile matching to a series of state-of-art approaches. Section 2 gives a brief introduction of components in song recommendation systems and in section 3, the state-of art recommendation techniques are explained. To the end of this paper, we conclude and propose a new model based on users motivation. The main objective of this system is 1.The system will determine the musical preferences of the users based on the analysis of their interaction during use. 2.This way the system is able to estimate what artist or group would match user preferences to the user at a given time. 3.These systems use information filtering techniques to process information and provide the user with potentially more relevant items.

- **LITERATURE SURVEY**

This section of the paper will represent a literature review of the works that are similar to the presented work. The literature survey shows that a hybrid model is projected which mixes user-based cooperative filtering and item-based cooperative filtering by adding the anticipated ratings from every technique and multiplying them with a weight that comes with the accuracy of every technique alone. The approach advantages from the correlation between not solely users alone or things alone however from each at the same time. The analysis was conducted on movielens dataset. the selection of weights was thought of by victimization and adjusting mean absolute error. therefore the survey shows that the hybrid approach improves the information scantness drawback and therefore

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

the accuracy of the system effectively and with efficiency.

1. Pasquale Lops, Marco de Gemmis and Giovanni Semeraro. 2010 Content-based Recommender Systems.
2. Robin Burke. 2010, Hybrid Recommender Systems.
3. Luo Zhenghua. 2012, Association rule Mining for recommendation system on the book sale.
4. Gilbert Badaro, Hazem Hajj, Wassim El-Hajj, and Lama Nachman, 2013
5. collaborative filtering for Recommender System: User based and Item based CF.
6. Anil Poriya, Neev Patel, Tanvi Bhagat, and Rekha Sharma, 2014, Non-Personalized Recommender Systems and User-based cooperative Recommender Systems
7. Aanand Shankar Tiwari, Abhay Kumar and Asim Gopal Barman. 2014, Content Based Filtering Collaborative Filtering and Assosiation Rule Mining.

- **PROBLEM STATEMENT**

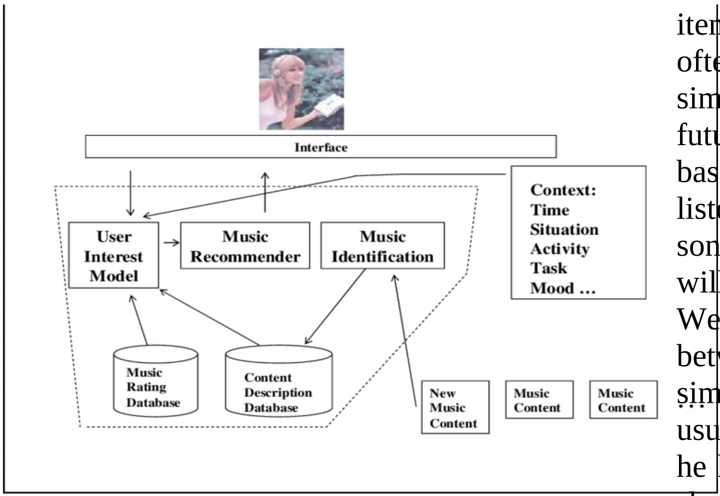
The basic task in song recommendation system with plagiarism detection is to generate the best music recommendation system by predicting based on customization and detecting the similar music genre to avoid copyrights issue, by using Collaborative filtering, Content based, Machine Learning, Data Analysis.

- **SCOPE**

Create a music recommendation system able to infer the userffs musical preferences in a given time. The scope is not to know the user; instead itffs about estimating what he could like right now. Explore the music services available nowadays looking for a complete and freely accessible music catalog and free streaming services. Develop a working system capable of making the most of free online services to provide the user with a completely free system which brings the opportunity of discovering new music.

- **METHODOLOGY**

4.1 System Architecture A system architecture or systems architecture is the conceptual model that defines the struc ture, behaviour, and more views of a system. An architecture description is a formal de scription and representation of a system, organized in a way that supports reasoning about the structures and behaviour structure of the system. it can consist of system components and the sub-systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture, collectively these are called architecture description languages(ADLs) Various organizations can define systems architecture in different ways, including:The fundamental organization of a system, embodied in its components, their relationships to each other and to the environment, and the principles governing its design and evolution. A representation of a system, including a mapping of functionality onto hardware and software components, a mapping of the software architecture onto the hardware architec ture, and human interaction with these components. An allocated arrangement of physical elements which provides the design solution for a consumer product or life-cycle process intended to satisfy the requirements of the functional architecture and the requirements baseline. An architecture consists of the most important, pervasive, top-level, strategic inventions, decisions, and their associated rationales about the overall structure (i.e., essential elements and their relationships) and associated characteristics and behavior. One can think of system architecture as a set of representations of an existing (or future) system. These representations initially describe a general, high-level functional organization, and are progressively refined to more detailed and concrete descriptions.



- **ALGORITHM/STEPS**

Steps:

- Start.
- Register yourself using Username, Email ID, Passward.
- After Registration Login yourself using correct credential like Username and Passward.
- Enter the Artist name.
- Listen your recommendate songs with its lyrics.
- Logout.

We have implemented four different algorithms to build an efficient recommendation system.

Popularity based Model:

It is the most basic and simple algorithm. We find the popularity of each song by looking into the training set and calculating the number of users who had listened to this song. Songs are then sorted in the descending order of their popularity. For each user, we recommend top most popular songs except those already in his profile. This method involves no personalization and some songs may never be listened in future.

Collaborative based Model:

Collaborative filtering involves collecting information from many users and then making predictions based on some similarity measures between users and between items. This can be classified into user-based and item based models. In item-based model, it is assumed that songs that are often listened together by some users tend to be similar and are more likely to be listened together in future also by some other user. According to user based similarity model, users who have similar listening histories, i.e., have listened to the same songs in the past tend to have similar interests and will probably listen to the same songs in future too. We need some similarity measure to compare between two songs or between two users. Cosine similarity weighs each of the users equally which is usually not the case. User should be weighted less if he has shown interests to many variety of items (it shows that either she does not discern between songs based on their quality, or just likes to explore). Likewise, user is weighted more if listens to very limited set of songs. The similarity measure, Wij = P(i/j), also has drawbacks that some songs which are listened more by users have higher similarity values not because they are similar and listened together but because they are more popular. We have used conditional probability based model of similarity[2] between users and between items: Wu,v = P(v/u) ffP(v/u) 1−ff, ff(0, 1) Wuv = |I(u) T I(v)| |I(u)| ff|I(v)| 1−ff [2] Different values of ff were tested to finally come with a good similarity measure. Then, for each new user u and song i, user-based scoring function is calculated as h U ui = P vui f(wuv)[2] Similarly, item-based scoring function is h I ui = P jIi f(wij)[2] Locality of scoring function is also necessary to emphasize items that are more similar. We have used exponential function to determine locality. f(w) = w q , qN[2] This determines how individual scoring components affects the overall scoring between two items. The similar things are emphasized more while less similar ones contribution drop down to zero. After computing user-based and item-based lists, we used stochastic aggregation to combine them. This is one by randomly choosing one of them according to their probability distribution and then recommending top scored items from them. When the song history of a user is too small to utilize the user-based recommendation algorithm, we can offer recommendations based on song similarity, which yields better results when number of songs is smaller than that of users. We got the best results for stochastic aggregation of item-based model with values of q and ff as 3 and 0.15, and of user-based model with values 0.3 and 5 for ff and q respectively, giving overall mAP 0.08212. This method does not include any personalization. Moreover, majority of songs have too few listeners so they are least likely to be recommended. But still, we as well as Kaggle winner got best results from this method among many others. Here, we have not used play count information in final result as they did not give good result because similarity model is biased to a few songs played multiple times and calculation noise was generated by a few very popular songs.

SVD Model:

Listening histories are influenced by a set of factors specific to the domain (e.g. genre, artist). These factors are in general not at all obvious and we need to infer those so called latent factors[4] from the data. Users and songs are characterized by latent factors. Here, to handle such a large amount of data, we build a sparse matrix[6] from user-song triplets and directly operate on the matrix, instead of looping over millions of songs and users. We used truncated SVD for dimensionality reduction. We used SVD algorithm in this model as follows: Firstly, we decompose Matrix M into latent feature space that relates user to songs. M = U PV , where M Rm∗n, Um∗k , Pk∗k and V k∗n Here, U represents user factors and V represents item factors. Then, for each user, a personalized recommendation is givn by ranking following item for each song as follows: Wi = U T u .Vi Though the theory behind SVD is quite compelling, there is not enough data for the algorithm to arrive at a good prediction. The median number of songs in a users play count history is fourteen to fifteen; this sparseness does not allow the SVD objective function to converge to a global optimum.

KNN Model:

In this method, we utilize the available metadata. We create a space of songs according to their features from metadata and find out neighborhood of each song. We choose some of the available features (e.g., loudness, genre, mode, etc.) which we found most relevant to distinguish a song from others. After creating the feature space, to recommend songs to the users, we look at each users profile and suggest songs which are neighbors to the songs present in his listening history. We have taken top 50 neighbors of each song. This model is quite personalized and uses metadata. But since, we had 280GB file of metadata which takes huge amount of time in processing, we extracted features of only 3GB (10,000 songs), which is less than 2 percent of total number. Due to this, we had features of only small number of songs, which gives us very small precision. SSBT's College of Engineering and Technology, Bambhori, Jalgaon (MS) 34

Evaluation Metrics:

We used mean Average Precision (mAP) as our evaluation metric. The reason behind using this is that this metric was used in the Kaggle challenge which helps us to compare our results with others. Moreover, precision is much more important than recall because false positives can lead to a poor user experience. Our metric gives the average of the proportion of correct recommendations giving more weight to the top recommendations. There are three steps of computing mAP as follows: Firstly, precision at each k is calculated. It gives the proportion of correct recommendations within the top-k of the predicted rankings. Pk(u, r) = 1 k Pk j=1 M(u, r(j)) Then, for each user, average precision at each k is evaluated. AP(u, r) = 1 nu Pt k=1 Pk(u, r).M(u, r(k)) Finally, mean of all the users is taken. mAP = 1 m Pm u=1 AP(u, ru)

- **RESULTS**

This assignment provided us with a fantastic learning opportunity. We've studied data mining and data cleansing.The music supplier can forecast and then give acceptable songs to its consumers using a song recommender system based on the qualities of previously heard music. for each song listened to by the user, the average vector of audio and metadata attributes Find the n-closest data points in the dataset (excluding points from the user's listening history) to this average vector. Take these n points and come up with some tunes to go with them. A research on the limits of an interactive music recommendation service based on artificial audio similarity calculation was provided. A number of computer experiments, as well as a review of real download data, reveal that a large chunk of the audio collection is only never or never suggested. A number of computer experiments are used to investigate this issue in depth, including the investigation of various audio similarity functions and comparisons with real download data. Our music recommendation service uses Gaussian mixtures as statistical models to determine timbre similarity. This is the de facto standard method for computing audiosimilarity, and it is recognised to produce high-quality results. A machine learning model's first goal is to eliminate all problem-causing objects from the dataset. Data cleansing and exploration were quite beneficial in getting the dataset algorithm ready. We learned how to design a machine learning model, train it, and then test it.

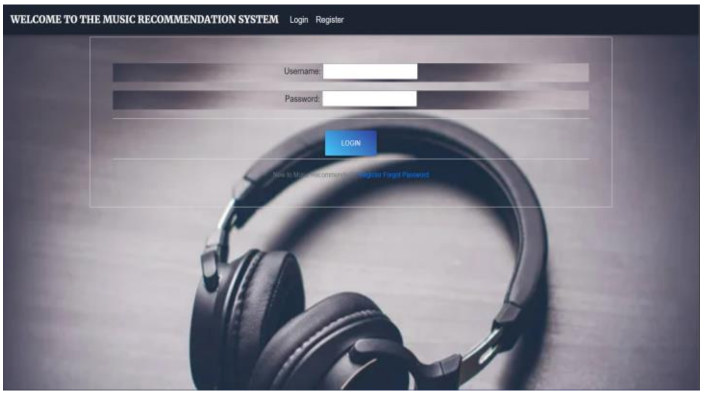


Figure : Login Page

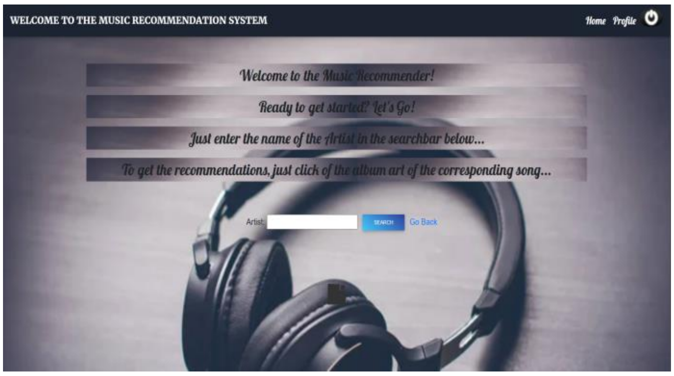


Figure : Home Page

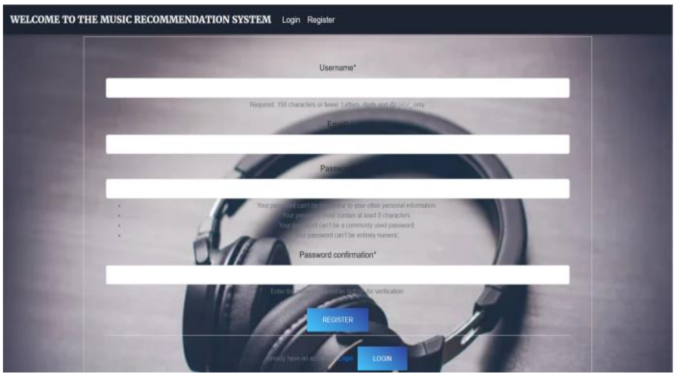


Figure : Register Page

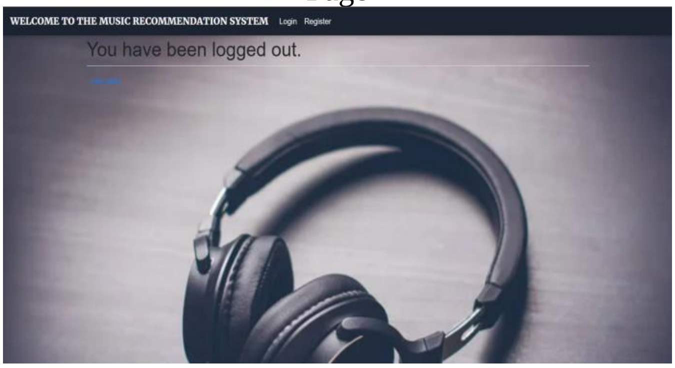


Figure : Login Page

- **CONCLUSION**

The following are our conclusions based on experiment results. First, song recommender system should consider the song genre information to increase the quality of song recom mendations. Second, CRNNs that considers both the frequency features and time sequence patterns has overall better performance. It indicates the effectiveness of its hybrid structure to extract the music features. Based on our analyses, we can suggest for future research to add other music features in order to improve the accuracy of the recommender system, such as using tempo gram for capturing local tempo at a certain time. Create a music recommendation system able to infer the userffs musical preferences in a given time. The scope is not to know the user; instead itffs about estimating what he could like right now.-Explore the music services available nowadays looking for a complete and freely accessible music catalog and free streaming services.-Develop a working system capable of making the most of free online services to provide the user with a completely free system which brings the opportunity of discovering new music.

- **REFERENCE**

1. Aiolli, F. (2012). A preliminary study on a recommender system for the million songs dataset challenge. PREFERENCE LEARNING: PROBLEMS AND APPLICATIONS IN AI
2. Spotify. (2013, March) Spotify Music for everyone. [Online].
3. https://www.spotify.com/us/ [10] (2013, March) Wikipedia. [Online].
4. Peter Knees and Markus Schedl. Music retrieval and recommendation – a tutorial overview. In Proceedings of the 38th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Santiago, Chile, August 2015.
5. Peter Knees and Markus Schedl. Music Similarity and Retrieval: An Introduction to Audioand Web based Strategies. 2016.
6. Chris Johnson. From idea to execution: Spotify’s discover weekly. November 2015.
7. Zhiwei Gu Li Guo and Tianchi Liu. Music genre classification via machine learning.
8. [11] Vikas Sindhwani Prem Melville, ”Recommender Systems,” in Encyclopedia of Machine Learning. Yorktown Heights, USA: IBM T. J. Watson Research Center, 2010, ch. 00338.
9. Beatriz C. F. de Azevedo Glaucia M. Bressan and Elisangela Ap. S. Lizzi.