IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

HarmonyAI: Intelligent Music Recommendations through Machine Learning

MR. KINGSLY STEPHEN[1],DEEPTHI.S[2], KRUPA.D[3], DEBAPRIYA SENGUPTA[4], PADALA PAWAN KUMAR[5].

[1]Assistant Professor, Department of Computer Science and Engineering ,SRMIST Ramapuram Chennai.

[2]Department of Computer Science and Engineering, SRMIST Ramapuram,Chennai.[UG Scholar]

[3]Department of Computer Science and Engineering, SRMIST Ramapuram,Chennai.[UG Scholar]

[4]Department of Computer Science and Engineering, SRMIST Ramapuram,Chennai.[UG Scholar]

[5]Department of Computer Science and Engineering, SRMIST Ramapuram,Chennai.[UG Scholar]

1.ABSTRACT:-

Existing techniques in user-individualized recommendation scenarios have faced significant challenges due to the vast amount of music data and the wide variety of listening habits. Many prior music recommendation algorithms only considered temporal links between records of sequential listening, ignoring the use of other data like the artist and album. A person's mood can be controlled, and profound emotional experiences can be evoked through music. The study of music recommendation services has advanced significantly since the introduction of online streaming services. The importance of information retrieved from lyrics and acoustic content has been disregarded by contemporary methods that leverage users' listening histories to inform session-based song suggestions. We approach song prediction using a variety of modalities, such as tags, lyrics, and acoustic content. We suggest a novel approach to ensemble learning for acoustic characteristics in this study. By combining poetic and acoustic data, our model outperforms current state-of-the-art models in terms of performance. Additionally, we conduct research to determine how users' psychological well-being affects the effectiveness of our models. Particularly, a piece of music frequently originates from and is part of an album by a particular musician. Singer and album information, which is regarded as music metadata, can be used as crucial supplementary information between various music pieces and may greatly affect the user's musical preferences. In this paper, we concentrate on the sequential recommendation task for music while considering additional information, and we propose a novel Graph based Attentive Sequential model with Metadata (GASM) that uses metadata to enhance music representations and efficiently mine user listening behaviour patterns. To be more specific, we first model the relationships between various types of nodes (user, music, singer, album) using a directed listening graph, and then we utilise graph neural networks to learn their latent representation vectors. The user's musical preference is then divided into long- term, short-term, and dynamic components using customised attention networks. Finally, GASM combines three different sorts of preferences to anticipate the upcoming (new) song in line with the user's musical interests. Three real-world datasets have been the subject of extensive testing, and the findings indicate that the suggested method, GASM, performs better than baselines. From a long list of available songs, choosing one that matches a user's preferences can be challenging. The primary problem with music software recommendations is that they correspond to song recommendations. Although this might not seem to be a problem at first, any typical music recommendation engine will undoubtedly notice a pattern and provide a monotonous selection of music, disappointing the user and making them lose interest. To keep the recommendations exciting and appealing to the user's tastes, our programme overcomes this problem by mixing in fresh music from different genres that matches the user's interests. Using machine learning and KNN, we present a customised music recommendation system in this work. We suggest a collaborative filtering and content filtering recommendation algorithm for the personalised music recommendation system that combines the output of the network with the log files to recommend music to the user. User data from the log file is used by the suggested technique to generate song recommendations for each recommendation. While collaborative approaches forecast potential preferences using a matrix with scores on various songs, content-based approaches make recommendations based on the similarities between the contents or attributes of two songs.

Keywords:- KNN, Machine learning, Collaborative filtering , GASM

2.INTRODUCTION:-

Due to the incredible advancements in information technologies over the past few decades, a vast amount of data has been stored in the database via the Internet. Information overload—a condition in which people are unable to get the facts they require in a timely manner—has thus grown to be a serious issue. As a result, recommender systems were developed to address this problem and have since been validated in numerous real-world contexts, including ecommerce, social networking sites, POI applications, and so on. A constantly growing and immensely popular art form, music has even become an integral part of many people's daily life. Numerous multinational corporations and online music services have emerged because of the thriving music industries, including Time Warner, Sony, PolyGram, EMI Group, iTunes, and NetEase Cloud Music. It can be quite difficult for most media platforms to choose the right songs for a user based on their listening history among millions of songs, especially when the user's interest and profile are unclear and there are few interaction records. As a result, the recommender system acts as a potent instrument to capture consumers' individual preferences even when confronted with a vast amount of data that is available, which can address the pressing issue of the music recommendation scene.

3.RELATED WORK:-

There has been extensive research done in the field of audio processing. One crucial step in audio processing is audio feature extraction. Some of the traditional audio features include Mel-Frequency Cepstral Coefficient (MFCC) [19], constant-Q chromagram [20], tempogram [21], etc. In recent years, researchers have also explored end-to-end audio feature extraction methods.

Collaborative methods utilize users' histories and assume that users with similar histories share similar interests. For example, Zhang et al. [25] propose a model called "Auralist" that is capable to consider four factors ("the desired goals of accuracy", "diversity", "novelty" and "serendipity") simultaneously. Other factors, such as tags and text descriptions are also exploited. For example, Sergio et al. [26] generate knowledge graphs for music recommendations.

Content-based methods extract information from the audio tracks directly. For example, Oord et al. [52] propose a deep convolutional neural networks based music recommendation method and the method outperforms a bag-of-words representation on the Million Song Dataset.

4.METHODOLOGY:-

There are only a few choices for developing a music recommendation system. EMP is a strong cross-platform music player with suggestion capabilities. EMP provides intelligent mood-based music suggestions by incorporating emotion context reasoning capabilities into the adaptive music recommendation engine. A music player contains three modules: the Emotion Module, the Music Classification Module, and the Recommendation Module. Deep learning techniques are used by the Emotion Module to accurately (90.23%) assess the user's mood from a face image. With an outstanding accuracy rate of 97.69%, the Music Classification Module employs audio information to categorize songs into four different mood groups.The recommendation module suggests songs to the user by matching the user's tastes and sentiments to the mood type of the music.

5.PROPOSED METHODOLOGY:-

The architecture exhibition at GASM is divided into four main sections:-
1. The creation of a music listening graph containing information
2. The learning of user and item representations
3. The capture of personal preferences
4. Music recommendation.

5.1 Construction Of a Music Listening Graph Using Metadata:-

We consider a user's listening to a certain piece of music to be their interactions with both the music and its information. The listening graph's user-item edges each represent a user interaction with a specific item. There are thus three different sorts of user-item edges: user-music, user-singer, and user-album. The music-music edges show a temporal link from listening records. The information about the album and artist serves as extra information for our main goal, which is the duty of making music recommendations. To improve the connection between metadata in the event of missing data, edges from the nodes of albums and singers point to the music nodes and there are bi-directional edges between albums and singers. Figure 4 provides an illustration of our method for creating the graph of music consumption

5.2 Learning From User and Item Representations:-

After creating the music listening graph, we employ GNN to get the user and item latent vectors. A low-dimensional vector, v R d, can be used to represent each node in the graph, where d specifies the size of the hidden vector's dimensions.

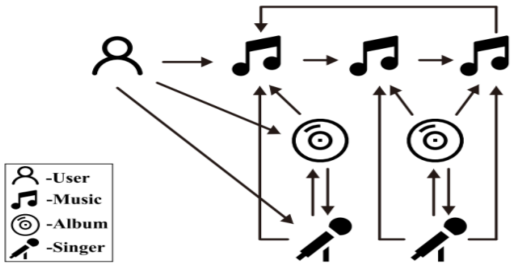
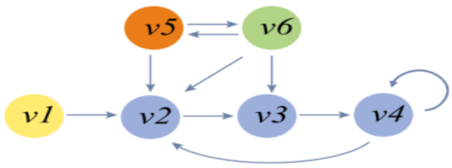


FIGURE 1:- Music Listening Graph with Metadata

The following definitions define the rules for updating each node in the GV,E:

$$
\begin{aligned}
\mathbf{a}_{s,i}^t &= \mathbf{A}_{s,i}^u : [\mathbf{v}_1^{t-1}, \ldots, \mathbf{v}_n^{t-1}]^\top \mathbf{H} + \mathbf{b}, \\
\mathbf{z}_{s,i}^t &= \sigma(\mathbf{W}_z \mathbf{a}_{s,i}^t + \mathbf{U}_z \mathbf{v}_i^{t-1}), \\
\mathbf{r}_{s,i}^t &= \sigma(\mathbf{W}_r \mathbf{a}_{s,i}^t + \mathbf{U}_r \mathbf{v}_i^{t-1}), \\
\widetilde{\mathbf{v}_l^t} &= \tanh(\mathbf{W}_o \mathbf{a}_{s,i}^t + \mathbf{U}_o(\mathbf{r}_{s,i}^t \odot \mathbf{v}_i^{t-1})), \\
\mathbf{v}_i^t &= (1 - \mathbf{z}_{s,u}^t) \odot \mathbf{v}_i^{t-1} + \mathbf{z}_{s,i}^t \odot \widetilde{\mathbf{v_l}}^t,
\end{aligned}
$$



Outgoing Edges

	v1	v2	v3	v4	v5	v6
v1	0	1	0	0	0	0
v2	0	0	1	0	0	0
v3	0	0	0	1	0	0
v4	0	1/2	0	1/2	0	0
v5	0	1/2	0	0	0	1/2
v6	0	1/3	1/3	0	1/3	0

Incoming Edges

	v1	v2	v3	v4	v5	v6
v1	0	0	0	0	0	0
v2	1/4	0	0	1/4	1/4	1/4
v3	0	1/2	0	0	0	1/2
v4	0	0	1/2	1/2	0	0
v5	0	0	0	0	0	1
v6	0	0	0	0	1	0

FIGURE 2:- A connection matrix and directed graph example.

5.2.1 Long-Term Preference:-

The user's consistent taste for music is represented by the long-term preference in GASM. To adjust to shifts in musical tastes or listener preferences as well as transitions in listening circumstances, we extract users' most recent listening history to predict their short-term preference. The most recent music a user heard has a substantial bearing on their future actions and provides important contextual information; hence, it should be given more weight. As a result, we construct the dynamic preference to deal with the uncertainty in listening behaviors by treating the final piece of music and its information.

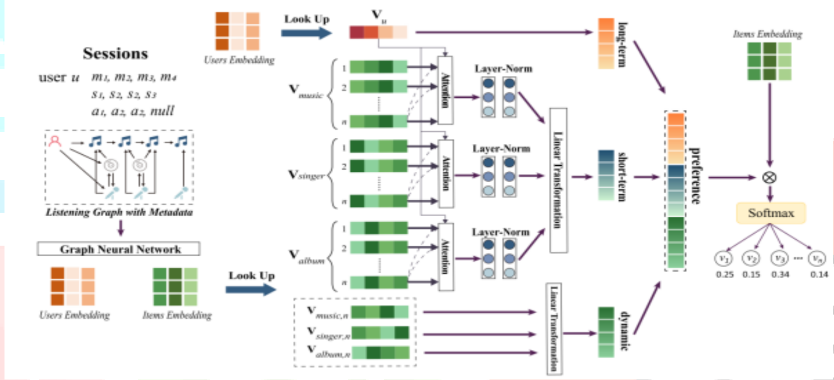


FIGURE 3:- GASM Framework

5.2.2 Short-Term Preference:-

To determine the short-term preference, we consider the n most recently played tracks in a session and use the attention mechanism to determine their various effects on upcoming listening behavior. Music, album, and singer item sequences are three distinct types of item sequences that are separately calculated and then concatenated via a linear transformation.

$$
\begin{aligned}
\mathbf{h}^{s}_{m,i} &= \phi(\mathbf{W}^{s}_{m,1}\mathbf{v}_{m,i} + \mathbf{W}^{s}_{m,2}\mathbf{v}_{m,n} + \mathbf{b}^{s}_{m}),\\
\alpha^{s}_{u,i} &= \frac{\exp(\mathbf{v}_u^{\top}\mathbf{h}^{s}_{m,i})}{\sum_{i\in U}\exp(\mathbf{v}_u^{\top}\mathbf{h}^{s}_{m,i})},\\
\mathbf{p}^{s}_{u,m} &= \sum_{i\in U}\alpha^{s}_{u,i}\mathbf{v}_{m,i},\\
\mathbf{p}^{s}_{u,m} &= \mathrm{layernorm}(\mathbf{p}^{s}_{u,m}),
\end{aligned}
$$

Additionally, we employ the layer-norm method to improve the stability of the training process and ensure the data distribution. Similarly, by using the following formulas, we can determine our short-term preferences for an album and a singer:

$$\begin{aligned}
\mathbf{h}_{a,i}^{s} &= \phi(\mathbf{W}_{a,1}^{s}\mathbf{v}_{a,i} + \mathbf{W}_{a,2}^{s}\mathbf{v}_{a,n} + \mathbf{b}_{a}^{s}),\\
\alpha_{u,i}^{s} &= \frac{\exp(\mathbf{v}_u^{\top}\mathbf{h}_{a,i}^{s})}{\sum_{i\in U}\exp(\mathbf{v}_u^{\top}\mathbf{h}_{a,i}^{s})},\\
\mathbf{p}_{u,a}^{s} &= \sum_{i\in U}\alpha_{u,i}^{s}\mathbf{v}_{a,i},\\
\mathbf{p}_{u,a}^{s} &= \mathrm{layernorm}(\mathbf{p}_{u,a}^{s}),
\end{aligned}$$

where Ws a,1 and Ws a,2 are the album weight matrices for a session.

$$\begin{aligned}
\mathbf{h}_{s,i}^{s} &= \phi(\mathbf{W}_{s,1}^{s}v_{s,i} + \mathbf{W}_{s,2}^{s}v_{s,n} + \mathbf{b}_{s}^{s}),\\
\alpha_{u,i}^{s} &= \frac{\exp(\mathbf{v}_u^{\top}\mathbf{h}_{s,i}^{s})}{\sum_{i\in U}\exp(\mathbf{v}_u^{\top}\mathbf{h}_{s,i}^{s})},\\
\mathbf{p}_{u,s}^{s} &= \sum_{i\in U}\alpha_{u,i}^{s}\mathbf{v}_{s,i},\\
\mathbf{p}_{u,s}^{s} &= \mathrm{layernorm}(\mathbf{p}_{u,s}^{s}),
\end{aligned}$$

Then, to determine the user's full short-term preference, we combine the user's interest in music, albums, and singers.

$$\mathbf{p}_u^{s} = \mathbf{W}_1[\mathbf{p}_{u,m}^{s} : \mathbf{p}_{u,a}^{s} : \mathbf{p}_{u,s}^{s}],$$

5.2.3 Dynamic Preference:-

The synthesis of the most recently listened to music and metadata, which is described as:

$$\mathbf{p}_u^{d} = \mathbf{W}_2[\mathbf{v}_{m,n} : \mathbf{v}_{a,n} : \mathbf{v}_{s,n}],$$

where W2 is the matrix for the linear transformation. There are distinct individual aspects in the user's listening behaviour that create a great deal of uncertainty when modelling the user's preference. To produce superior recommendations, we carefully consider long-term, short-term, and dynamic components and immediately concatenate them to create the user's hybrid preference Pu

$$\mathbf{p}_u = \mathbf{W}_3[\mathbf{p}_u^{l} : \mathbf{p}_u^{s} : \mathbf{p}_u^{d}],$$

W3 is the linear transformation matrix in this instance.

5.3 Random Forest Classification:-

A well-liked machine learning technique called andom Forest is frequently employed for classification and regression problems. We shall emphasise Random Forest classification in our response. Multiple decision trees are combined to make predictions using the ensemble learning technique known as Random Forest classification. A random subset of the training data and a random subset of the characteristics are used to separately construct each decision tree in a Random Forest. In order to maximise the separation between the classes, a decision tree is constructed by recursively dividing the data into smaller groups based on the values of the input features. Each decision tree in a Random Forest is trained using a distinct subset of the data and characteristics, making them diverse and vulnerable to many causes of mistake. However, we can obtain a more precise and reliable prediction by combining the predictions of all the trees. In order to produce predictions, the Random Forest algorithm typically takes the mean (for regression) or the mode (for classification) of each forecast from each tree.

5.1 Advantages of Random Forest Algorithm:-

- Because the combination of many trees lessens the risk of individual trees overfitting to the data, Random Forests are robust to noise and overfitting.
- Large datasets with numerous features and classes can be handled using Random Forests.
- Estimates of each feature's contribution to the classification can be provided using Random Forests, which is helpful for choosing features.
- Random Forests are scalable to huge datasets and are simple to parallelize.
- Large datasets are efficiently handled by it.
- It retains accuracy even when a sizable amount of the data is missing thanks to an efficient mechanism for guessing missing data.
- As the forest grows, it produces an internal, unbiased estimate of the generalization error.
- Less memory is needed, and less work is required in the computation. Uses neither density nor distance measurements to find abnormalities. This reduces the significant computing cost of distance calculation in all density- and distance-based approaches.
- Decreased computing times because anomalies are quickly and early detected.
- Functions when superfluous features are added.

6.CONCLUSION:-

We consider it to be fantastic since it provided an opportunity for us to put the theories we had learnt in the course into reality, to apply them, and to try to better understand a real-world AI challenge: the Music Recommender System. We gain in-depth knowledge of several approaches to this problem, particularly the four models that we have discussed in the study. By updating the dataset, the learning set, and the testing set, changing different issue settings, and evaluating the results, we obtain a lot of practicing skills. We encountered numerous difficulties managing this huge dataset, figuring out how to better examine it, and resolving some technical quirks. With a lot of effort, we have managed to overcome all of these. The teamwork on this project has been outstanding. We both come from different cultures and have different ways of doing things due of this. It took some time for us to become accustomed to one another, to settle in, and to function as a team. We become significantly more productive and have more fun as the team spirit grows. Both of us have enjoyed working on this project, so it was worth the effort. We have learned several new things because of this initiative. To improve our studies, we still have a lot of research to complete. Given how wide, uncharted, and challenging the field of music recommender systems is, we can take some efforts and carry out various experiments in the future. We realized that developing a recommender system is not a simple task. Because of the scale of the dataset, it presents several challenges. First off, choosing 500 accurate songs from a pool of 380 million songs requires a high degree of precision, which is not an easy task. This prevented us from getting any results higher than 10%. Even the Kaggle winner only received 17%. Second, it can be difficult to separate the information in the metadata that is relevant to a track while studying it. Thirdly, it technically takes a lot of CPU and memory resources to process such a large dataset. As a result of all these problems with the data and the system itself, it is both harder and more alluring. We anticipate having more chances to work in artificial intelligence in the future. We are sure that we can do better.

REFERENCES:-

[1]. McFee, B., BertinMahieux, T., Ellis, D. P., Lanck riet, G. R. (2012, April). The million song dataset challenge . In Proceedings of the 21st international conference companion on world wide web (pp. 909916. ACM .

[2]. Aiolli, F. (2012). A preliminary study on a recom mender system for the million songs dataset chat lenge. PREFERENCE LEARNING: PROBLEMS AND APPLIATIONS IN AI.

[3]. Korean , Yehuda. “Recommender system utilizing col liberative filtering combining explicit and implicit feedback with both neighbourhood and factor models .

[4]. Cremonesi , Paolo, Yehuda Korean , and Robert Tur rin. “Performance of recommender Algorithms on top n recommendation tasks .” Proceedings of the fourth ACM Conference on Recommender systems. ACM, 2010

[5]. T. Bertin et al., The Million song dataset, Proc. Of the 12th international society for Music information Retrieval Conference, 2011

[6]. Labrosa.ce.columbia.edu, (2017). Million Song Datasetscaling MIR research. [online] Available at: https://labrosa.ee.columbia.edu/millionsong [Accessed10Oct. 2017].

[7]. Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nic. Xia Hu, and Tat-Seng Chua. 2017. NeuralCollaborative Filtering In Proceedings of the 26th InternationalConference on World Wide Web (WWW '17). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva,
Switzerland, 173-182. DOI: https://doi.org/10.1145/3038912.3052569

[8]. Min SH., Han L. (2005) Recommender Systems Using SupportVector Machines. In: Lowe D., Gaedke M. (eds) Web Engineering. ICWE 2005. Lecture Notes in ComputerScience, vol3579. Springer, Berlin, Heidelberg