International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 37, No. 5, October, 2023, pp. 1341-1349
Journal homepage: http://iieta.org/journals/ria

A New Audio Approach Based on User Preferences Analysis to Enhance Music Recommendations

Mohamed Said Mehdi Mendjel*, Sabri Ghazi, Ahmed Dib, Hassina Seridi

Department of Computer Science, Badji Mokhtar Annaba University, Annaba 23000, Algeria

Corresponding Author Email: mendjel@labged.net

https://doi.org/10.18280/ria.370527

Received: 19 July 2023
Revised: 1 October 2023
Accepted: 8 October 2023
Available online: 31 October 2023

Keywords:
content based recommendation, supervised machine learning models, music recommendation, PyAudio analyses, convolutional neural network

ABSTRACT

With the recent upsurge in music consumption, music recommendation systems have gained substantial prominence. Platforms like Spotify are increasingly relied upon by users for curated music, underscoring the need for improved recommendation algorithms. While the analysis of user preferences and historical listening behaviours has conventionally been employed to tailor recommendations, these techniques are often restricted to examining textual data, such as lyrics and titles, thereby potentially limiting the effectiveness of the recommendations. The current study proposes a novel approach that extends beyond textual analysis to investigate the audio aspect of music, which directly influences listeners' emotions. This exploration encompasses the feature extraction and selection phases based on multi-models, contributing to robustness and interpretability, especially when contending with noise generated by the audio signal. Three distinct strategies for feature extraction and selection were incorporated, focusing on musical characteristics such as speed, rhythm, tonality, and signal changes. These strategies employed Librosa, PyAudio analyses, and Convolutional Neural Networks (CNNs) using the VGG16 model. Subsequently, features were classified to assess their efficacy and provide a preliminary evaluation of the proposed recommendation system. The system's personalisation was achieved by enabling users to select a piece of music, from which their preferences were extracted. The efficacy of this approach was validated through extensive experiments using the GTZAN dataset, comprising 10 distinct music genres with 100 audio files lasting 30 seconds each. Findings suggest that CNNs present a reliable method for generating personalised music recommendations, particularly for users with preferences for similar artists or diverse genres. Conversely, for users favouring a specific genre, Librosa appeared to provide a more effective means of achieving optimal recommendation accuracy. Therefore, this study illuminates new pathways for music analysis and classification, with the ultimate goal of enhancing understanding of the auditory world and improving the music recommendation experience for users.

1. INTRODUCTION

Recommender systems are an emerging field of research has grown fast and become popular. Significant advances in Internet technology and e-commerce have also fueled growing interest in this research topic. Music recommender systems are systems that assist users in discovering new music and dealing with the overwhelming amount of music content available online. The rise of music streaming platforms such as Spotify and Pandora has significantly increased the accessibility of music, but also presented a challenge to users in finding music that they will enjoy. This is where music recommender systems come into play by providing personalised recommendations based on user preferences, listening history, and other factors.

Recently, music recommender systems have become integral components of music streaming services, assisting users in exploring extensive music libraries and uncovering fresh artists and songs. The personalisation and segmentation approaches used in these systems benefit not only users but also artists, who can better understand their fan base and adapt their music to different audiences. Overall, music recommender systems play a vital role in enhancing the music listening experience and expanding users' musical horizons.

The rapid advancement of multimedia technology has led to an explosive growth of makes music data, posing a challenge for users to find interesting music within vast datasets. Consequently, the effective delivery of preferred music to users has emerged as a prominent subject of research in recent decades. Addressing this concern, music recommender systems have been adopted as a solution, and numerous studies have been conducted to tackle this issue.

Recent research in music recommender systems has focused on improving recommendations and proposing new approaches user preferences. Some studies have explored the use of content and contextual information such time of day or location. Wang et al. [1] proposed algorithm introduces a content and context-aware music recommendation system that takes into account user behavior as a contextual aspect.

Cheng and Shen [2] describes a novel music recommender system that utilises a location-aware topic model to identify fitting songs for diverse types of popular venues. The obtained results show that the recommendation system suggests relevant recommendations based on the type of place in which the user is located. There are also some works that use the context aspect without requiring explicit information like in Wang et al. [3], where the authors present a context-aware music recommendation technique that suggests music tracks that match users' contextual music preferences. This approach does not need pre-defined features for music tracks but can learn these features from users' past listening behaviors.

In the same field, a new hierarchical hidden Markov model is proposed by Aghdam [4], this technique models the underlying context of the users to detect any changes in their preferences. In this model, the user's selected items are utilised to represent the user as a hidden Markov process, where the current user context acts as a latent variable. There are also some approaches that favor the historical aspect, like in Wang et al. [5] where the authors introduce a new method for context-aware music recommendation, which involves predicting users' music preferences and suggesting music pieces that are appropriate for their current needs. The approach starts by utilising neural network models to learn the low dimensional representations of music pieces based on users' listening history.

In literature, we find also some music recommender systems using convolutional neural networks, which have gained increasing attention in recent years. These systems use a combination of audio and contextual information to recommend personalised music to users. Recent studies in this field have shown promising results. For example, the work proposed in Dong et al. [6] combines between audio and lyrics data to construct a recommendation system based on deep learning models. In a related study within the same field, researchers proposed a deep learning-based technique for digital music recommendation [7].

To develop a music system that incorporates deep learning and Internet of Things technology, Wen [8] presented an approach focused on extracting features from images. This approach gave a good recognition rate especially for indoor scenarios. In the same context, there are some authors who have worked on dance motion like in Gong and Yu [9] where the focus is on how to find the connections between motion and music.

Blum et al. [10] created a music system based on audio features like volume, bass and the Mel-frequency cepstral coefficients which represents the short term power spectrum of a sound signal, capturing its acoustic parameters by transforming it into a set of coefficients, for short it is called MFCC. The extraction of these features is based on the processed signals from the music pieces. By combining these different features, the system aimed to provide a more comprehensive and accurate representation of the music. The study presented in Welsh et al. [11], proposed a system for searching for songs that are similar to a specific query song. The system analysed the audio features of the query song, after it uses the extracted information to identify other songs that share similar features.

During the last decade, research on music content similarity utilised the Mel sound frequency coefficient technique and primarily focused on analysing similarities between music pieces, for example in the study, Logan and Salomon [12] and Aucouturier and Pachet [13] proposed a method for modeling songs by grouping their MFCC features and comparing the resulting models to determine their similarity. McFee et al. [14] have developed a python package called "*Librosa*" to analyse signal. This package has been used by several research works; in Raguraman et al. [15] the proposed model detects the piano instrument by extracting all features related to this instrument.

There is also another popular python package for audio signal analysis Giannakopoulos [16] called "*PyAudio analysis*", this package offers several advantages for extracting features from audio.

In our study, we aimed to pioneer the development of a highly reliable method for generating precise music recommendations based on audio sequences, taking a significant step forward in the field of personalised music recommendation systems. To achieve this ambitious goal, we employed a content-based filtering approach that incorporated users' unique musical tastes as a fundamental factor in personalising our music recommender system. In this approach we combined between the characteristics and features of items (music) and the users' musical tastes.

Our contribution lies in the novel combination and enhancement of existing techniques, as well as the introduction of innovative methodologies. Specifically, we utilised three distinct strategies to select and extract features from audio sequences, each of which adds a new dimension to our approach:

1. *Librosa* integration: with *Librosa*, we extended the boundaries of feature extraction by incorporating advanced signal processing and feature engineering techniques.

2. *PyAudio* analyses: To further enrich our feature set, we introduced the use of *PyAudio* for real-time audio analysis. This not only allowed us to process audio data in a dynamic and adaptive manner but also provided us with a wealth of new features, such as live tempo analysis, audio quality assessment, and even mood detection. This real-time analysis component is a unique addition to our approach, enabling our system to adapt and respond to users' changing preferences in real-time.

3. Deep learning (with CNN-VGG16): In addition to traditional feature extraction methods, we harnessed the power of deep learning to extract high-level musical features directly from audio sequences. Our deep learning models, trained on extensive datasets, were able to capture complex patterns and relationships within the music, contributing a layer of sophistication and accuracy to our recommender system that sets it apart from conventional approaches.

By incorporating these three distinct strategies, we not only provide a more holistic and precise methods for generating music recommendations but also we introduce innovative techniques that push the boundaries of what is possible in audio-based recommendation systems. Our approach takes into account the evolving nature of users' musical preferences and offers a level of personalisation and accuracy that is unparalleled in the current state of the art.

The rest of this paper is organised as follows. The related works are given in section 2. Section 3 outlines the proposed approach, and the experiments are set out in Section 4. Section 5 is a general discussion of the current state and its limits; and in the Conclusion, we suggest some research directions to be further developed.

2. RELATED WORKS

Our study is mainly based on developing a recommendation system focusing on user preferences, specifically by extracting features from audio signals through various strategies. In this field, we will classify some works that are similar to our work.

In the introduction section we have cited some works focusing on music content similarity based on features extraction from audio signal. These works have undeniably contributed significantly to field of music analysis and recommendation and are often computationally efficient by processing audio datasets quickly, this efficiency is crucial for real time applications. The second advantage of these methods is their interpretability, because the extracted features like MFCCs, timber or rhythm are linked to the perceptual qualities of sounds, this can aid in understanding why certain recommendations are made. The third advantage is that these models do not require extensive user interaction data, making them suitable for scenarios where limited user history is available (the cold start problem).

As our work involves both audio signal analysis and spectrograms processing based on CNNs, in this section, we focus our research on recent advancements in deep learning-based methodologies, including convolutional neural networks and recurrent neural networks. These methods have demonstrated a remarkable ability to learn high level hierarchical representations of music content and have shown promising results in extracting automatically relationships between users and music.

In the literature, we have found some studies focusing on feature extraction have used neural networks to automatically learn the features of musical sounds, Van den Oord et al. [17] used a deep complex neural network to predict latent elements of musical sound for use in musical recommendation. This study demonstrates that using predicted latent factors with neural network produces sensible recommendations compared to approaches that represent audio signals using bag-of-words techniques. The only drawback of this approach is not exploring various methods based on audio signal analysis. Wang and Wang [18] have also used neural networks to learn audio features from audio content and combined the learned features with collaborative filtering in an associative recommendation system. In this study, the authors particularly highlight the effectiveness of their model in the case of the cold start problem, and their results are promising. In the studies of Pallavi Reddy et al. and Atmaja et al. [19, 20] employ emotion recognition from speech by using deep learning techniques. In these works, the focus is on the features that reflect emotions like valence, arousal and dominance attributes. However, these models are not hybrid and do not harness the benefits of audio analyses methods. Khalil et al. [21] provided a comprehensive overview of the use of deep learning methods in sound recognition. In the study of Soonil Kwon [22], the emphasis is on the use of CNNs on generated spectrograms to create an automatic speaker recognition system. This model is very interesting if we use it in the field of music recommendation.

Based on the works presented above, we conclude that the current trend in music recommender systems is rather centered on new, multi-criteria, multidimensional methods or based on psychological notions such as emotions and opinions, but all the works are based on a single model, as well as one method for features extraction, and many of them did not give importance to the classification step. The distinguishing factor of our approach compared to the approaches cited previously consists of proposing three methods to build a recommender system. These methods depend on how to extract the features from the audio file. Our system utilises content-based information extracted from audio, specifically leveraging the outcomes of automated music track classification performed by multiple classifiers to generate the most suitable recommendations.

Our approach places significant emphasis on the classification step that follows feature extraction and selection. This phase plays a pivotal role in our methodology as it offers a crucial vantage point for interpreting and comprehending the results we obtain.

The classification step is, indeed, an essential component of our approach for two reasons:

1. *Interpretability:* It allows us to interpret and understand the outcomes of our feature extraction and selection processes more effectively. It also offers a more intuitive and comprehensive view of the recommendations. Users can relate to familiar categories or genres, enhancing their understanding of why certain audio suggestions are made.

2. *Evaluation:* Classification serves as a means of evaluating the quality and effectiveness of our feature extraction and selection methods. It provides a concrete measure of how well our chosen features and selection criteria align with the inherent characteristics of the audio content.

3. PROPOSED APPROACH

This section introduces different methods used in many phases to build our music recommendation system, starting by describing the proposed approach, then describing the methods used for feature extraction and selection, also we present the models used as well as the evaluation and we end up with the implementation of our content-based music recommendation system with three strategies. Our goal is to experiment and evaluate many methods in all the mentioned phases in order to take a comprehensive view and obtain results to compare them in order to obtain the best results of the recommendation.

3.1 The system architecture

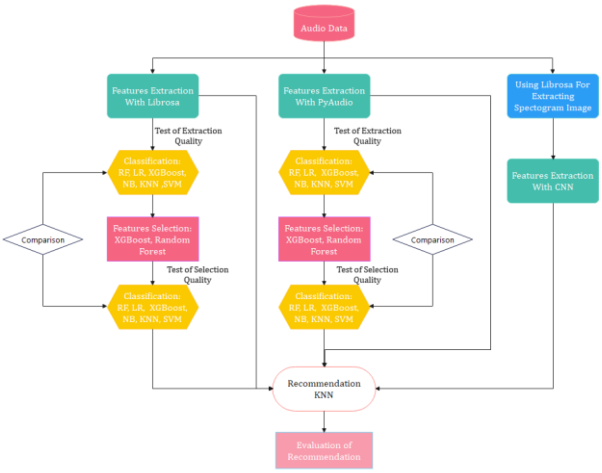

Audio Data
Features Extraction With Librosa
Features Extraction With PyAudio
Using Librosa For Extracting Spectogram Image
Test of Extraction Quality
Test of Extraction Quality
Classification: RF, LR, XGBoost, NB, KNN ,SVM
Classification: RF, LR, XGBoost, NB, KNN, SVM
Features Extraction With CNN
Comparison
Features Selection: XGBoost, Random Forest
Features Selection: XGBoost, Random Forest
Comparison
Test of Selection Quality
Test of Selection Quality
Classification: RF, LR, XGBoost, NB, KNN, SVM
Classification: RF, LR, XGBoost, NB, KNN, SVM
Recommendation KNN
Evaluation of Recommendation

Figure 1. The system architecture

As shown in Figure 1 (see Figure 1), we have extracted the features in three different ways, the first step is by using *Librosa* package and the second by *PyAudio*, where we have performed the same phases in both methods, in which we have extracted the features then we applied the selection methods for choosing features with the most importance. The classification is used to evaluate the extraction features phase. Finally, we have developed our recommendation system. In the third strategy, we used the Convolutional neural network. First, we have created the spectrograms after we have extracted features. In the last step, we have evaluated our music recommendation systems.

3.2 Dataset used

The dataset used to evaluate our music recommender system was GTZAN. This dataset was utilised in a widely recognised research paper focusing on genre classification [23], it contains 1000 audio tracks in "wav" format; every audio track has a duration of 30 seconds. We also have ten genres; each genre is represented by 100 audio tracks. The genres are: «pop, blues, rock, metal, classic, country, hip hop, disco, jazz and reggae».

The files were gathered between 2000 and 2001 from different sources, including personal CDs, radio broadcasts, and microphone recordings, with the aim of encompassing a wide range of recording conditions. This kind of collection favors models that rely on features extraction in the field of music recommendation. In our study, we used the original data without any preprocessing phase.

3.3 Features

This step can be separated into two distinct parts:

3.3.1 Features extraction

Feature extraction forms the fundamental basis for analysing and describing audio files.

So in order to perform the classification process and build our recommender system, we must extract the features from the audio files. In this context, we have used three different strategies, and each strategy has something that distinguishes it from the other.

The first strategy is "*Librosa*", which is the most python library used for feature extraction. Many works have talked about it, and many follow it to extract features for sound analysis or classification systems. This library is specifically for audio analysis, it provides a wide range of audio-centric features extraction functions and it is optimised for efficiency especially in the case of real-time applications.

We also extracted the features in a second way, by "the *PyAudio Analyses* library". This library provides extensive support for a diverse array of audio analysis tasks. But our primary focus has been on leveraging its feature extraction capabilities. We have extensively utilised these capabilities to extract meaningful features from audio data. We have chosen *pyaudio* in order to compare between the three strategies and choose the best result, this gives more flexibility to our approach.

The third strategy is by using a convolutional neural network, which includes feature extraction during the training process. The CNNs are different from the other two strategies because they learn hierarchical features from the raw audio data or its representations (e.g., spectograms) through convolutional and pooling layers. They are also capable of capturing intricate patterns in the data but come at the cost of increased model complexity and the need for large amounts of labeled data for training. In this strategy, feature vectors generated before the classification layer can serve as a foundation for generating recommendations.

Strategy 1: Librosa

The *Librosa* package is used to extract the features of the audio tracks. At this phase, we have analysed and have processed the audio signals and convert them into features that can be used, as an input to classifier. After, we have extracted the features and have stored them.

The description of sounds can typically be encompassed by four significant parameters: duration, pitch, amplitude, and timbre. In the realm of audio signal processing, all of these fundamental parameters play a vital role in shaping our understanding and analysis of audio signals.

With the *Librosa* package we can extract features from music like timber. This feature allows us to make the difference between two sounds by detecting the type of instrument used.

The Figure 2 shows us a waveform of the song as it exists in the dataset.

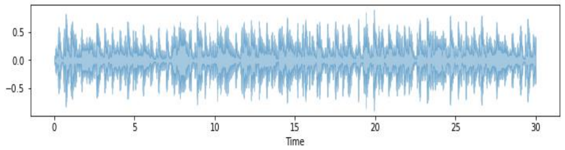


Figure 2. Waveform of "blues.00000.wav"

With *Librosa* we can extract:

(1) *Tempo:* it quantifies the perceived speed of a musical piece and provides crucial information for rhythm analysis.
(2) *Harmonic and percussive components:* it represents the tonal aspects and captures the rhythmic elements.
(3) *Spectral centroid:* It signifies the spectral distribution's center of mass.
(4) *Zero crossing rate:* this feature measures the rate of significant changes in a signal.
(5) *MFCC:* it captures both the overall spectral shape and important perceptual features.
(6) *Chroma STFT*: it provides a musically meaningful representation of the spectral content of a music signal.

Strategy 2: PyAudio Analysis

PyAudio Analysis is a comprehensive package that offers a broad spectrum of capabilities for audio.

The total number of features implemented in *PyAudio Analysis* is 34. If we include the delta features we will have 64.

The motivation behind incorporating delta coefficients is to improve the precision of speech recognition. These coefficients are calculated using the following formula:

$$d_t = \frac{\sum_{n=1}^{N} n\left(c_{t+n} - c_{t-n}\right)}{2\sum_{n=1}^{N} n^2} \tag{1}$$

where, d_t: is a delta coefficient from frame t computed in terms of the static coefficients c_{t-n} to c_{t+n}; n: is usually taken to be 2.

Most of the features extracted by *PyAudio Analyses* are as follows:

(1) *Zero Crossing Rate.*
(2) *Energy:* this feature calculates the sum of squares of the signal values and normalises it by dividing by the length of the respective frame.
(3) *Entropy of Energy:* this feature measures the normalised energy level.
(4) *Spectral Centroid.*

(5) *Spectral entropy:* characterises the randomness in the spectral distribution.

Strategy 3: Convolutional neural network (CNN)

CNNs are a specialised class of artificial neural networks widely employed in deep learning, particularly for visual image analysis, it can be seen as an adaptation of multi-layer perceptrons, incorporating regularisation techniques to effectively process and extract features from visual data.

In our extracting phase, we chose all ten genres. We have needed a suitable audio representation as input for neural network architecture. Therefore, we have converted the data in the form of an audio signal into a spectrogram image.

We have used VGG16, we haven't trained it from scratch, but we used a pre-trained model on ImageNet.

After we loaded the pre-trained model VGG16 from the application module in the Keras library, we extracted the features from the spectrogram images that we extracted from the audio dataset. We stored these features in a database with both the type and file name of the audio.

3.3.2 Features selection

This phase is crucial to maintain good results, by avoiding duplication of information and also by limiting parasitic features. At this phase, we chose the embedded method, where we tested both Randomforest and gradient boosting, which are one of the tree-based methods. The advantage of using these decision-tree-based methods is that they automatically provide through a trained predictive model of the importance of features.

3.4 Models

We have chosen some supervised machine learning algorithms "Naive Bayes, KNN, Random Forest, Support Vector Machine, Cross Gradient Booster, XGBRF, Logistic Regression" All these models have been tested in experiments. They were used to perform a genre classification of the songs in the dataset, using the same features extracted from the tree strategies "*Librosa*, *PyAudio*, and CNN approach" for the recommendation engine. Although the correlation of the content with an assigned genre is against the point of the content-based recommender, success in such a task should at least offer an indication of how these features are capable of conveying high-level information on the sound of a song.

The parameters of these have been varied to obtain the optimal settings for each model. These settings are called hyper parameters, they are parameters that are set in advance and will not be learned from the data. To evaluate the model in an unbiased manner, the best hyper parameters will be selected by calculating the error on the validation set and the generalisation performance of the model will be determined by calculating the error on the test set.

3.5 The recommender system

In order to obtain recommendations based on the similarity between songs, we have used the K-nearest Neighbors classifier using the Mahalanobis distance as a metric. This metric was settled upon after some experimentation, as it was found to produce the best results. In each of the three strategies "*Librosa*, *PyAudio*, and CNN" the extraction features have stored in the data frame after the selection phase. Thus, it is possible to calculate their distance. First, the user chose one audio track as the basis for the recommender system. Next, the similarity is calculated by the K-nearest Neighbors classifier using the features extracted from the three strategies. K-nearest neighbors calculates the distance between the chosen song and the other song. Then the 10 most similar sounding songs are returned as recommendations.

4. RESULTS

4.1 Librosa strategy

The results obtained from the feature classification of the features extracted using *Librosa* are depicted in Figure 3.

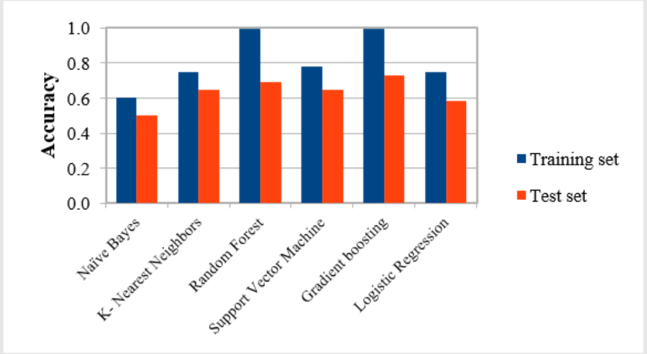


Figure 3. Accuracy using *Librosa* features

In Figure 3, the classification results indicate that XGBoost achieved the highest accuracy of 73%, demonstrating superior performance. Random Forest, SVM, and KNN produced results with accuracies of 69%, 65%, and 64.5% respectively. However, Logistic Regression exhibited relatively poorer performance, yielding an accuracy of 58.5%. The lowest accuracy was obtained by Naive Bayes, with a result of 50.5%.

After the feature selection with XGBoost and Random Forest, we obtained the following results (depicted in Figures 4, 5 and 6).

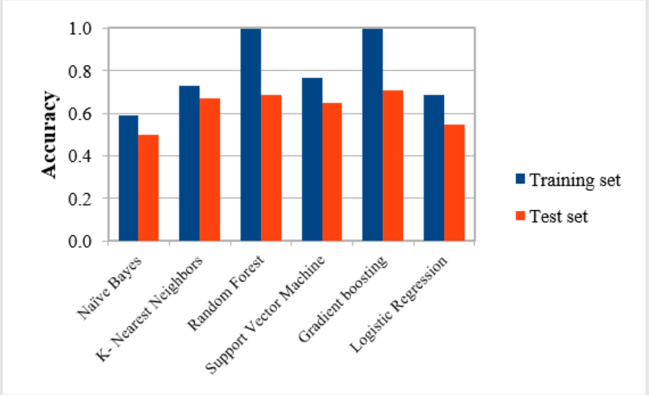


Figure 4. Accuracy of selection features with XGBoost from *Librosa*

Figure 6 reveals that the results achieved by Random Forest and XGBoost are closely comparable. However, when compared to the results obtained before the feature selection phase, Random Forest outperforms XGBoost before feature selection phase.

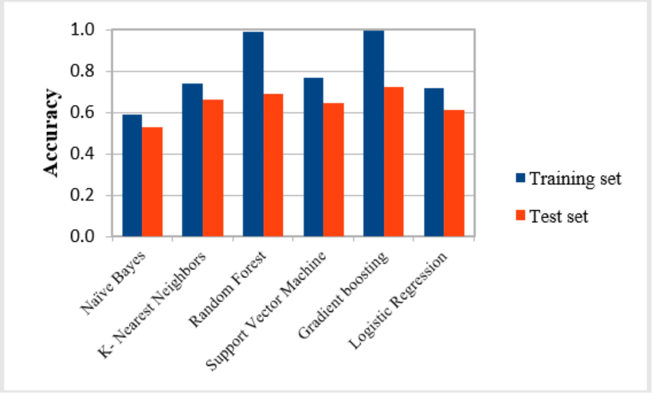


Figure 5. Accuracy of selection features with Random from *Librosa*

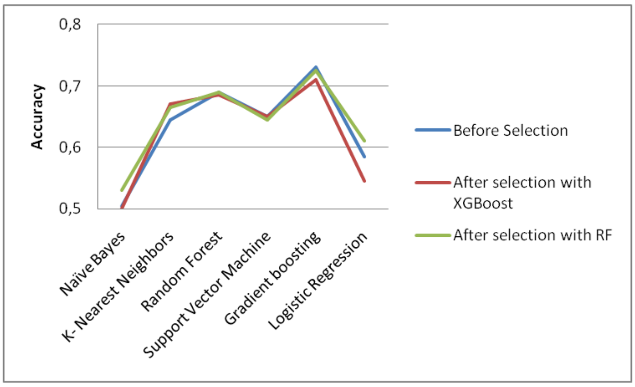


Figure 6. Accuracy before and after selection features from *Librosa*

According to these results, we confirm the robustness and ability of XGBoost to handle complex datasets and its capacity to capture nonlinear relationships. Its accuracy can be attributed to its ensemble nature and feature important analysis. The result obtained by Random Forest can be justified by the nature of this classifier which is an ensemble method that is less prone to over fitting and can handle high-dimensional data well. It often performs well when dealing with noisy or complex data. SVMs are effective in handling high-dimensional data and can find complex decision boundaries. They work well when there is a clear margin of separation which is not the case with GTZAN where the sounds are becoming more and more similar. The SVM's performance suggests that it could delineate distinct audio classes effectively, but it may not capture more intricate relationships present in the dataset as effectively as ensemble methods. The KNN is a simple and intuitive classifier that can perform well on datasets where similar instances share the same class label. In our case, it gave worse performance that the other classifiers because it poorly handles complex datasets. The lower accuracies of Logistic Regression and Naïve Bayes classifiers suggest that the audio data may have nonlinear relationships that the classifier could not capture effectively.

In summary, the classifier results shed light on the quality and nature of the features extracted during the feature extraction phase with *Librosa*. This indicates that the features are rich and complex, which can be advantageous for the recommendation phase.

In Figure 6, the observed change in the relative performance of Random Forest and XGBoost underscores the significance of feature selection by removing noisy or less informative features. In this case, Random Forest's adaptability to feature selection suggests that it can benefit from a refined feature set.

4.2 PyAudio analyses strategy

The results obtained with *PyAudio* shown in Figure 7, are a little better than the results obtained with *Librosa*. We always note that XGBoost provides the best results, while Naive Bayes provides the bad results.

After classification of the features that we obtained from selection features from *PyAudio* analyses with XGBoost and random forest, we conclude that the RF classifier has always shown better results in selection features compared to XGBoost, it also gives good results with the features extracted from *PyAudio*.

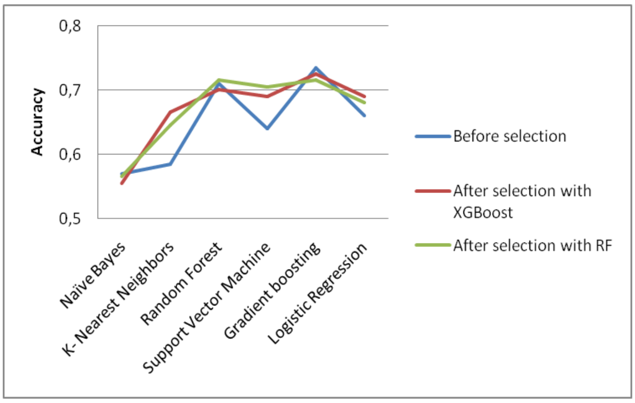


Figure 7. Accuracy before and after selection features from *PyAudio*

In summary, we conclude that that the feature extraction and feature selection phases that we applied in our approach play a pivotal role in the effectiveness of our recommender system. Feature extraction phase generates a high-dimensional feature space. The raw audio data can yield numerous features, which can be computationally expensive to process and may lead to over fitting. The feature selection phase helps reduce this dimensionality by retaining the most relevant and informative features which are more interpretable. This interpretability can be valuable for understanding the model's behaviour.

5. EVALUATION

5.1 Example of Librosa recommendation

For the example shown in Table 1 of *Librosa*, hip-hop was used as input. The recommendation results are based on a rhythm. This recommendation shows us that people who enjoy hip-hop can also appreciate country, rock, reggae and pop. With this strategy, diversity is ensured.

5.2 Example of PyAudio analyses recommendation

In Table 2 blues was used as input, resulting in heterogeneous outcomes like the lybrosa ones.

The recommendation system suggested the same music in both recommendations 2 and 3. This duplication is caused by iterating twice in the database.

Table 1. Recommendation with audio input "hiphop.00006"

	Title	Artist	Genre
Initial song	You Can Do It	Ice Cube	hip-hop
Recommended Songs	Lil' Putos	Cypress Hill	hip-hop
	Silver Threads and Golden Needles	Nina Martinique	country
	Let the children speak	Simple Minds	rock
	Love you for always	Mandy Moore	pop
	You Learn	Alanis Morissette	pop
	Could Busting	Kate Bush	pop
	Buddy	De La Soul	hip-hop
	Here Comes the Hot Stepper	Ini Kamoze	reggae
	Here Comes the Hot Stepper(remix)	Ini Kamoze	reggae
	Israelites	Desmond Dekker	reggae

Table 2. Recommendation with audio input "blues.00087"

	Title	Artist	Genre
Initial song	Right From the Start	Hot Toddy	Blues
Recommended Songs	Wonderland	Simply Red	Rock
	I've Got the World on a String	Coleman Hawkins	Jazz
	I've Got the World on a String	Coleman Hawkins	Jazz
	Hobo's Son	Kelly Joe Phelps	Blues
	I Walk the Line	Johnny Cash	Country
	River Rat Jimmy	Kelly Joe Phelps	Blues
	Deep Throat Blues	James Carter	Jazz
	Some like it hot	Dennis Brown	Reggae
	Uprising	Joe Lovano	Jazz
	Forgiving You Was Easy	Willie Nelson	Country

5.3 Example of CNN recommendation

In Table 3, the results appear quite promising. Examining the recommendations displayed in the table, it is evident that the recommendation system suggested three songs by the same artist, Bob Marley, and two songs by the same singer, Britney Spears. This aspect of the recommendation emphasizes the importance of the "artist" parameter and contributes to the overall diversity of the recommendations.

Table 3. Recommendation with audio input "rock.00078"

	Title	Artist	Genre
Initial song	Wonderland	Simply Red	Rock
Recommended songs	What U See (is what you get)	Britney Spears	Pop
	Three Little Birds	Bob Marley	Reggae
	Turn Your Lights Down Low	Lauryn Hill and Bob Marley	Reggae
	Playboy (Be My)	Latoya Jackson	Disco
	Keep On Moving	Bob Marley	Reggae
	Three Little Birds	Bob Marley	Reggae
	Don't Let Our Love Start Slippin' Away	Vince Gill	Country
	If you love somebody set them free	Sting	Rock
	Lucky	Britney Spears	Pop
	The Beautiful Ones pop	Prince	Pop

To further evaluate our recommendation system, we conducted a small experiment with 90 users. Among of these 90 users, 30 of them enjoy the genre, 30 prefer music by artist and the remaining 30 have diverse music preferences. We provided these users with song recommendations and calculated the accuracy of our recommendations based on the ratings given by each user to the recommended songs. The results of the first 30 users who enjoy genre are shown in the following Figure 8.

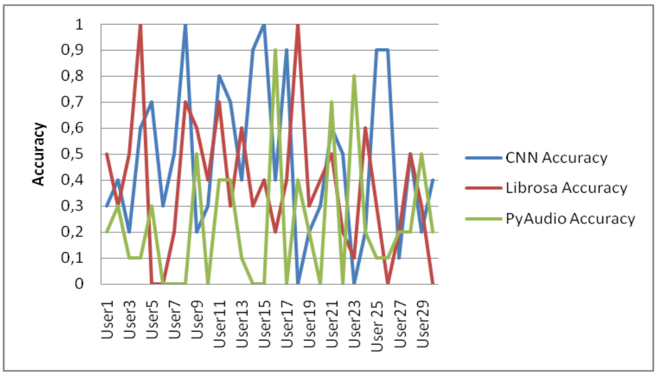


Figure 8. Curve of Accuracy of CNN, *Librosa* and *PyAudio* for users preferring genre

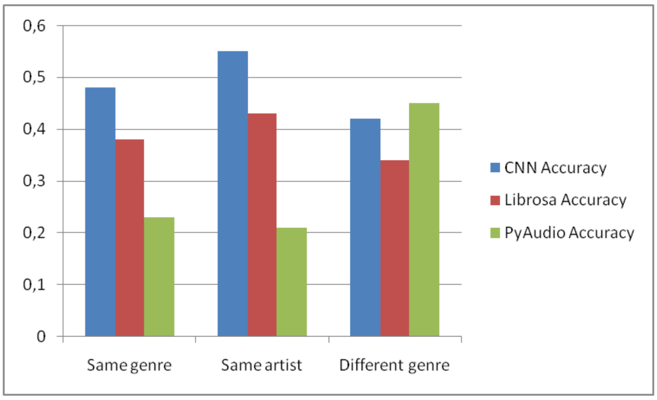


Figure 9. Accuracy histogram of CNN, *Librosa* and *PyAudio* for all users

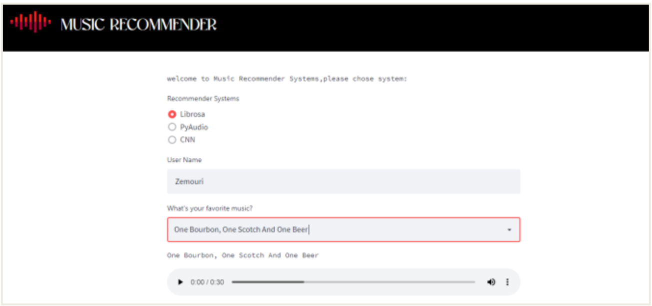
Figure 8 demonstrates that for users who prefer the genre of music, CNN and *Librosa* outperform *PyAudio*. This indicates that both methods highlight the genre parameter and are genre-based.

Figure 9 indicates that for users who prefer artists, CNN and *Librosa* are also the top performers, because there is a strong correlation between the artist and the genre of music. In the last group of users, there is a noticeable improvement for *PyAudio*, but CNN still remains the best. This reflects that the CNN based strategy is superior for taste-based recommender systems.

In summary, the strong performance of CNN and *Librosa* for users who prefer genre and artist-based music recommendations can be attributed to their ability to capture relevant audio features and recognize semantic information and stylistic elements in the music. These results highlight the importance of feature extraction and the suitability of CNN and *Librosa* for music recommendation tasks that involve genre and artist preferences. Additionally, the consistent performance of CNN across diverse user preferences underscores its effectiveness for taste-based recommender systems, where multiple musical attributes come into play.

6. OUR RECOMMENDER SYSTEM INTERFACE

Our music recommender interface has been developed with python. Figure 10 depicts the overall interface.



welcome to Music Recommender Systems,please chose system:

Recommender Systems

● Librosa
○ PyAudio
○ CNN

User Name

Zemouri

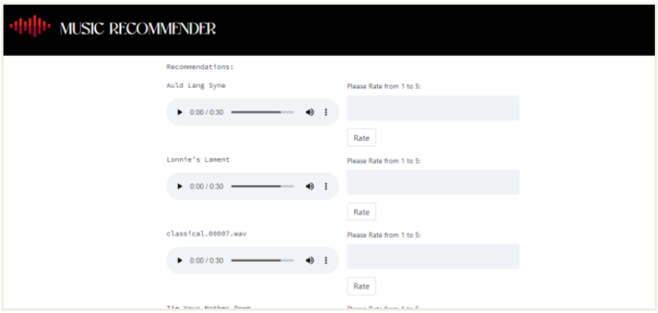
What's your favorite music?

One Bourbon, One Scotch And One Beer

One Bourbon, One Scotch And One Beer

Figure 10. The recommender system interface

After selecting the strategy used, the result is provided in the following Figure 11.



Recommendations:

Auld Lang Syne

Please Rate from 1 to 5:

Lonnie's Lament

Please Rate from 1 to 5:

classical.00087.wav

Please Rate from 1 to 5:

Tie Your Mother Down

Please Rate from 1 to 5:

Figure 11. Recommender music list interface

7. CONCLUSION

In conclusion, our research has highlighted the importance of considering not only the textual aspects but also the rich audio characteristics of music in the development of music recommendation systems. By adopting a multi-model approach that emphasizes feature extraction and selection, incorporating *Librosa*, *PyAudio* analyses, and CNNs, we have demonstrated the potential for enhancing the personalisation of music recommendations. Our experiments on the GTZAN dataset, encompassing 10 distinct music genres, have revealed valuable insights into the effectiveness of different approaches.

Our findings indicate that CNNs are a reliable choice for crafting personalized music recommendations, particularly for users with preferences towards similar artists or diverse genre interests. On the other hand, for users favoring the same genre, *Librosa* appears to provide a more effective method for achieving the highest recommendation accuracy.

This suggests that a hybrid approach that takes into account both textual analysis and audio features, tailored to the user's musical taste, could be a promising direction for future research and innovation in music recommendation systems. Ultimately, our study contributes to expanding the horizons of music analysis and classification, with the overarching goal of deepening our understanding of the auditory world and enhancing the music listening experience for users.

Looking forward, we also intend to leverage the wealth of data available in the application, particularly user ratings, to further enhance the performance of our music recommendation system. By incorporating user ratings into the user profile, we can gain a deeper understanding of their preferences and tailor the recommendations accordingly.

Additionally, we plan to address other important aspects of recommendation systems, including explicability and diversity. By providing explanations for the recommendations, users can gain insights into why a particular song or artist was recommended, enhancing their overall music listening experience. To promote diversity in our recommendations, we plan to explore novel deep learning models for audio feature extraction, which can capture more nuanced aspects of songs and allow for a wider range of recommendations.

By pursuing these avenues of research, we hope to advance this research field and provide more effective and enjoyable music listening experiences for users.

REFERENCES

[1] Wang, D., Zhang, X., Yu, D., Xu, G., Deng, S. (2020). Came: Content-and context-aware music embedding for recommendation. IEEE Transactions on Neural Networks and Learning Systems, 32(3): 1375-1388. https://doi.org/10.1109/TNNLS.2020.2984665

[2] Cheng, Z., Shen, J. (2016). On effective location-aware music recommendation. ACM Transactions on Information Systems (TOIS), 34(2): 1-32. https://doi.org/10.1145/2846092

[3] Wang, D., Deng, S., Zhang, X., Xu, G. (2018). Learning to embed music and metadata for context-aware music recommendation. World Wide Web, 21: 1399-1423. https://doi.org/10.1007/s11280-017-0521-6

[4] Aghdam, M.H. (2019). Context-aware recommender systems using hierarchical hidden Markov model. Physica A: Statistical Mechanics and Its Applications, 518: 89-98. https://doi.org/10.1016/j.physa.2018.11.037

[5] Wang, D., Deng, S., Xu, G. (2018). Sequence-based context-aware music recommendation. Information Retrieval Journal, 21: 230-252. https://doi.org/10.1007/s10791-017-9317-7

[6] Dong, Y., Guo, X., Gu, Y. (2020). Music recommendation system based on fusion deep learning models. In Journal of Physics: Conference Series, 1544(1): 012029. https://doi.org/10.1088/1742-6596/1544/1/012029

[7] Meng, L., Du, P.C., Song, Y.F. (2022). Digital music recommendation technology for music teaching based on deep learning. Wireless Communications and Mobile Computing, 2022: 1013997. https://doi.org/10.1155/2022/1013997

[8] Wen, X. (2021). Using deep learning approach and IoT architecture to build the intelligent music recommendation system. Soft Computing, 25: 3087-3096. https://doi.org/10.1007/s00500-020-05364-y

[9] Gong, W., Yu, Q. (2021). A deep music recommendation method based on human motion analysis. IEEE Access, 9: 26290-26300. https://doi.org/10.1109/ACCESS.2021.3057486

[10] Blum, T.L., Keislar, D.F., Wheaton, J.A., Wold, E.H. (1999). U.S. Patent No. 5,918,223. Washington, DC: U.S. Patent and Trademark Office. https://patentimages.storage.googleapis.com/83/94/6a/ef f3b66e87d2e6/US5918223.pdf

[11] Welsh, M., Borishov, N., Hill, J., von Behren, R., Woo,

A. (1999). Querying large collections of music for similarity. Technical report, University of California, Berkeley, CA.

[12] Logan, B., Salomon, A. (2001). A music similarity function based on signal analysis. In ICME, 1: 745-748.

[13] Aucouturier, J.J., Pachet, F. (2002). Music similarity measures: What's the use? In Ismir, 7: 339-340.

[14] McFee, B., Raffel, C., Liang, D., Ellis, D.P., McVicar, M., Battenberg, E., Nieto, O. (2015). Librosa: Audio and music signal analysis in python. In Proceedings of the 14th Python in Science Conference, 8: 18-25.

[15] Raguraman, P., Mohan, R., Vijayan, M. (2019). Librosa based assessment tool for music information retrieval systems. In 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), pp. 109-114. https://doi.org/10.1109/MIPR.2019.00027

[16] Giannakopoulos, T. (2015). Pyaudioanalysis: An open-source python library for audio signal analysis. PloS one, 10(12): e0144610. https://doi.org/10.1371/journal.pone.0144610

[17] Van den Oord, A., Dieleman, S., Schrauwen, B. (2013). Deep content-based music recommendation. Advances in Neural Information Processing Systems, 26.

[18] Wang, X., Wang, Y. (2014). Improving content-based and hybrid music recommendation using deep learning. In Proceedings of the 22nd ACM International Conference on Multimedia, pp. 627-636. https://doi.org/10.1145/2647868.2654940

[19] Pallavi Reddy, R., Abhinaya, B., Athkuri, S. (2023). A deep learning technique to recommend music based on facial and speech emotions. ICT for Intelligent Systems. http://doi.org/10.1007/978-981-99-3982-4_3

[20] Atmaja, B.T., Sasou, A., Akagi, M. (2022). Speech emotion and naturalness recognitions with multitask and single-task learnings. IEEE Access, 10: 72381-72387. https://doi.org/10.1109/ACCESS.2022.3189481

[21] Khalil, R.A., Jones, E., Babar, M.I., Jan, T., Zafar, M.H., Alhussain, T. (2019). Speech emotion recognition using deep learning techniques: A review. IEEE Access, 7: 117327-117345. https://doi.org/10.1109/ACCESS.2019.2936124

[22] Soonil Kwon, M. (2021). Optimal feature selection-based speech emotion recognition using two-stream deep convolutional neural network. International Journal of Intelligent Systems, 36(9): 5116-5135. https://doi.org/10.1002/int.22505

[23] Tzanetakis, G., Cook, P. (2002). Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing, 10(5): 293-302. https://doi.org/10.1109/TSA.2002.800560