Project Idea

**Developing a Music Recommendation System Using Spectrograms and Librosa to Improve Accuracy and Efficiency with User Feedback**

**Technology Stack:**

* Python
* Librosa
* Spotify API
* HTML
* CSS
* JavaScript

This project aims to develop a music recommendation system that uses spectrograms of songs from the Spotify API and compares them using the Python library Librosa to make the recommendation cluster more accurate. The front end of the website will be developed using HTML, CSS, and JavaScript. The Spotify API will be used to get music metadata and tracks.

**Project Description:**

The following are the key steps involved in the project:

1. Collecting a dataset of songs from the Spotify API: The first step is to collect a dataset of songs from the Spotify API. This can be done using the Spotify API's /search endpoint. The dataset should include a variety of songs from different genres and artists.
2. Narrowing down the process using music metadata: Before using spectrogram comparison, you can first narrow down the process using the existing system of music recommendation which uses music metadata. This will help to improve the efficiency and accuracy of the system. You can filter the songs by genre, artist, mood, and other music metadata to get a list of songs that are more likely to be relevant to the user.
3. Computing spectrograms of the songs: Once the dataset of songs has been collected and narrowed down, the next step is to compute spectrograms of the songs. This can be done using the Librosa library's stft() function.
4. Comparing the spectrograms using Librosa: Once the spectrograms of the songs have been computed, the next step is to compare them using Librosa. This can be done using the Librosa library's mfcd() function to compute the mel-frequency cepstral distance (MFCD) between the spectrograms. The MFCD is a measure of the similarity between two spectrograms.
5. Clustering the songs based on their MFCDs: Once the MFCDs between all of the songs have been computed, the next step is to cluster the songs based on their MFCDs. This can be done using a variety of clustering algorithms, such as k-means clustering or hierarchical clustering.
6. Recommending songs to users: Once the songs have been clustered, the next step is to recommend songs to users based on their listening history. This can be done by finding the clusters that are most similar to the user's listening history and then recommending songs from those clusters.
7. Using user feedback to improve the accuracy of the system over time: As users listen to the recommended songs and rate them, you can use this feedback to update the system's model and improve its recommendations.

The front end of the website will allow users to browse and search for songs. The website will also display recommendations to users based on their listening history.

The following are the benefits of using spectrograms and Librosa for music recommendation:

* **Spectrograms** capture the musical features of songs in a way that is independent of the audio format. This means that the music recommendation system can be used to recommend songs from a variety of sources, such as Spotify, YouTube, and SoundCloud.
* Librosa provides a number of functions for comparing spectrograms. This makes it easy to develop a music recommendation system that is accurate and efficient.
* Using user feedback to improve the accuracy of the system over time will make the system more personalized and relevant to each user.

The following are the challenges involved in the project:

* Collecting a large and diverse dataset of songs from the Spotify API. The dataset should include a variety of songs from different genres and artists in order for the music recommendation system to be accurate.
* Developing a clustering algorithm that is able to group songs together based on their MFCDs. The clustering algorithm should be able to identify clusters of songs that are musically similar.
* Developing a front end that is user-friendly and allows users to browse, search, and get recommendations for songs.
* Designing and implementing a user feedback loop that can be used to improve the accuracy of the system over time.

Overall, this project is a challenging but rewarding opportunity to develop a music recommendation system that is more accurate and efficient than existing systems. The project will contribute to the field of music information retrieval and provide users with a better way to discover new music.