Team: 150

MUSIC RECOMMENDATION SYSTEM

2103A52058 Mohammed Mudassir Hussain
2103A52069 Syed Khaja Mukarram Ajaz
2103A52159 Mohammed Naveed Sharief

Mentor and Trainer :
Dr. Mamta Pandey
Associate Professor, School of CS&AI

Abstract

This project develops a feature-driven Music Recommendation System using machine learning models and the Spotify API to suggest tracks based on intrinsic audio features like danceability, energy, and acousticness. Data was integrated from a Kaggle dataset and over 4,300 unique tracks sourced via the Spotify API, covering diverse genres such as Indie, Hip-Hop, Global Charts, and Regional Hits. Advanced algorithms, including K-Nearest Neighbors (KNN), Cosine Similarity, PCA with KNN, and DBSCAN, were implemented to evaluate performance in generating accurate recommendations. A user-friendly Streamlit interface enables real-time interaction, delivering dynamic and relevant music suggestions. This hybrid approach effectively combines content-based filtering with advanced clustering techniques, offering a scalable and adaptive solution for personalized music discovery.

Introduction

The rise of online streaming platforms like Spotify and Apple Music has revolutionized the digital music industry, offering users access to millions of tracks. However, this abundance of options often makes discovering new, relevant songs a challenge, driving the need for personalized music recommendation systems. Traditional approaches, such as collaborative and content-based filtering, face issues like cold-start problems and scalability with expanding song libraries, limiting their effectiveness.

This project addresses these challenges by developing a hybrid Music Recommendation System that combines content-based filtering with advanced machine learning models. Leveraging real-time data from the Spotify API and a Kaggle dataset, the system analyzes audio features such as danceability, energy, and acousticness to deliver personalized recommendations. A user-friendly Streamlit interface further enhances the experience by enabling real-time searches and scalable song additions, ensuring adaptability and improved user satisfaction.

Motivation

- With millions of tracks available on streaming platforms, discovering relevant music has become a significant challenge.
- Existing recommendation systems often rely on limited metadata, leading to repetitive and less diverse suggestions.
- Leveraging audio features such as danceability, tempo, and energy can improve the accuracy and diversity of recommendations.
- Real-time integration of data ensures the system remains relevant and adapts to the latest music trends.
- Building a scalable system that handles diverse musical styles addresses critical challenges like the cold-start problem, enhancing music discovery.

LITERATURE REVIEW

Core Methodologies in Music Recommendation

Collaborative Filtering (CF) analyzes user interaction data to recommend items based on user-item relationship patterns. While effective in capturing shared preferences, it struggles to personalize recommendations for new or less popular songs due to the cold-start problem. [1] *Content-Based Filtering (CB)* focuses on intrinsic audio features, such as tempo, energy, or genre, to recommend songs similar to those a user has interacted with. However, its lack of user-specific context and history limits adaptability to diverse user preferences.[2] *Hybrid Systems* combine CF and CB approaches to enhance recommendation diversity and accuracy. By leveraging both user behavior and song-specific features, they mitigate the individual limitations of CF and CB.[3]

LITERATURE REVIEW

Advanced Techniques and Our Approach

Emerging techniques such as tensor factorization and reinforcement learning have added layers of sophistication to music recommendations but often require significant computational resources. In contrast, our project adopts a hybrid approach, integrating CB with advanced clustering methods like KNN, PCA-KNN, and DBSCAN. This methodology, supported by real-time data fetched via the Spotify API, delivers dynamic, feature-rich recommendations. By addressing cold-start issues and scalability challenges, the system balances precision and adaptability while providing an intuitive user interface for seamless interaction.

Existing Method v/s Proposed Method

Traditional methods like collaborative filtering and basic content-based filtering face limitations such as the cold-start problem, lack of diversity, and inability to handle real-time data. Collaborative filtering relies on user interaction data, making it ineffective for new users or songs, while basic content-based filtering struggles to provide diverse recommendations.

Our proposed method enhances content-based filtering by integrating real-time data from the Spotify API and utilizing advanced machine learning models like KNN, Cosine Similarity, PCA with KNN, and DBSCAN Clustering. These techniques analyze audio features such as danceability, energy, and acousticness to deliver accurate, diverse, and personalized recommendations. With a Streamlit-based interface, the system ensures seamless interaction and scalability.

Existing Method v/s Proposed Method

Table 1: Existing v/s Proposed methods

Aspect	Existing Methods	Proposed Method
Data Source	Historical user interactions	Merged dataset (Kaggle + Spotify API)
Cold-Start Problem	Significant issue	Mitigated using real-time data fetching
Model Variety	Limited (collaborative/content)	KNN, Cosine Similarity, PCA, DBSCAN
Scalability	High computational demand	Efficient due to feature engineering
User Experience	Static recommendations	Dynamic updates and diverse suggestions

Existing Method v/s Proposed Method

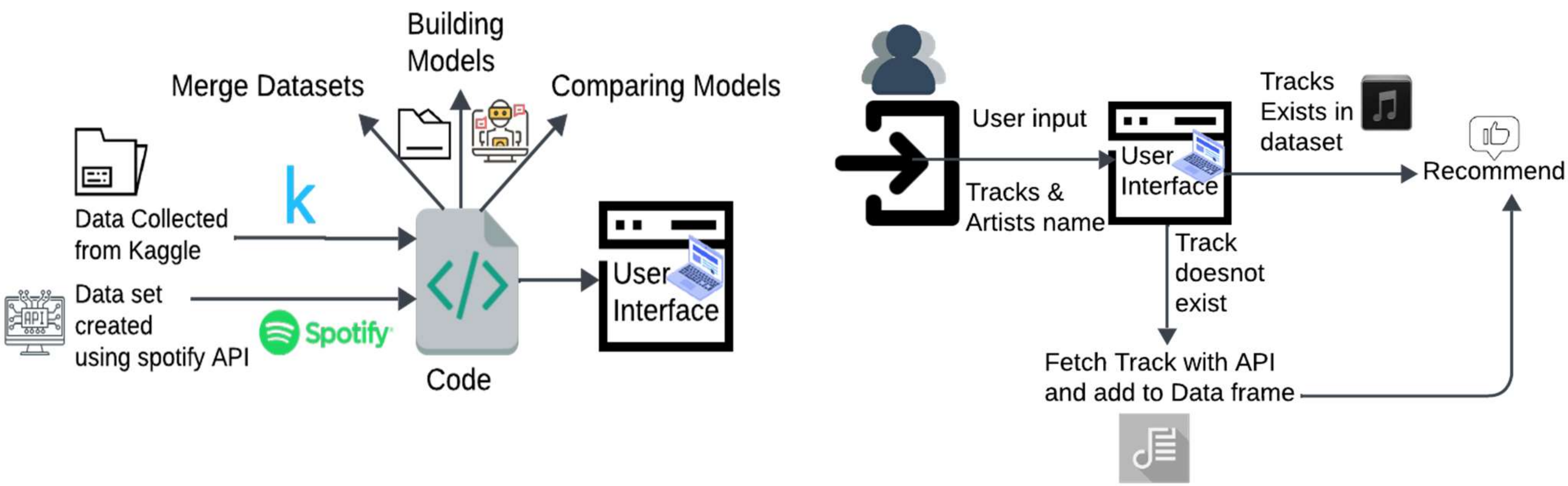

Figure 1: Proposed plan

OBJECTIVE

Primary Goal: To develop a personalized music recommendation system using machine learning models, focusing on analyzing audio features and providing real-time, accurate song recommendations.

Sub -Goals:

- Collect and preprocess music data from Spotify API and Kaggle datasets, focusing on audio features such as danceability, energy, acousticness, and tempo.
- Implement advanced content-based filtering models using KNN, Cosine Similarity, PCA with KNN, and DBSCAN Clustering to improve recommendation accuracy and diversity.
- Build a user-friendly interface using Streamlit to enable real-time song recommendations and seamless interaction for users.

DATASET

- Spotify API: Collected real-time data on 4,341 tracks across multiple genres.
- Kaggle Dataset: Historical data with audio features and metadata for over 600k songs.
- Features used include danceability, energy, tempo, acousticness, valence, and more.
- Combined datasets ensure a wide range of songs and styles for analysis.

DATASET

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	id	name	popularity	duration_r	explicit	artists	id_artists	release_da	danceabili	energy	key	loudness	mode	speechines	acousticne	instrument	liveness	valence	tempo	time_signature
2	35iwgR4jX	Carve	6	126903	0	['Uli']	['45tIt06X(	########	0.645	0.445	0	-13.338	1	0.451	0.674	0.744	0.151	0.127	104.851	3
3	021ht4sdg	CapÃ-tulo	0	98200	0	['Fernandc	['14jtPCOc	########	0.695	0.263	0	-22.136	1	0.957	0.797	0	0.148	0.655	102.009	1
4	07A5yehtS	Vivo para (	0	181640	0	['Ignacio C	['5LiOoJbx	########	0.434	0.177	1	-21.18	1	0.0512	0.994	0.0218	0.212	0.457	130.418	5
5	08FmqUhx	El Prisione	0	176907	0	['Ignacio C	['5LiOoJbx	########	0.321	0.0946	7	-27.961	1	0.0504	0.995	0.918	0.104	0.397	169.98	3

Figure 2: Dataset found on Kaggle

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	danceabili	energy	key	loudness	mode	speechines	acousticne	instrument	liveness	valence	tempo	type	id	uri	track_href	analysis_u	duration_r	time_signa	track_nam	cluster
2	0.425541	-1.45391	6	0.310759	0	-0.63853	1.471144	-0.35918	-0.68931	0.793963	-0.96658	audio_fea	3bmsaFp1	spotify:tra	https://api	https://api	171064	4	nadaaniya	0
3	-0.6762	1.031467	0	0.748808	1	-0.5496	-0.94341	-0.28401	-0.20603	-0.55372	-0.82272	audio_fea	5ivRSlOhV	spotify:tra	https://api	https://api	313163	4	Aaoge Tun	0
4	-0.36046	0.838838	6	-0.09472	1	0.53669	-0.85654	-0.35887	-0.74052	-0.1452	1.752585	audio_fea	0GamKm5	spotify:tra	https://api	https://api	174286	4	Bye	0
5	-0.42764	-1.16732	2	-0.78502	1	-0.52771	1.060739	-0.35807	0.682114	0.528638	1.970846	audio_fea	5Z2GQ0eB	spotify:tra	https://api	https://api	148967	4	Lamhey	0

Figure 3: Dataset saved after fetching from Spotify API

METHODOLOGY

- Audio features such as danceability, energy, tempo, and acousticness were collected from Kaggle and Spotify API datasets.
- The dataset was cleaned, preprocessed, and scaled to ensure consistency in feature values for analysis.
- Machine learning models, including K-Nearest Neighbors (KNN), Cosine Similarity, PCA with KNN, and DBSCAN Clustering, were implemented to identify similar tracks and group them effectively.
- Dynamic updates using the Spotify API addressed the cold-start problem, ensuring scalability and inclusion of new tracks.
- An interactive Streamlit-based interface allowed users to input song details and receive real-time music recommendations.

RESULTS:

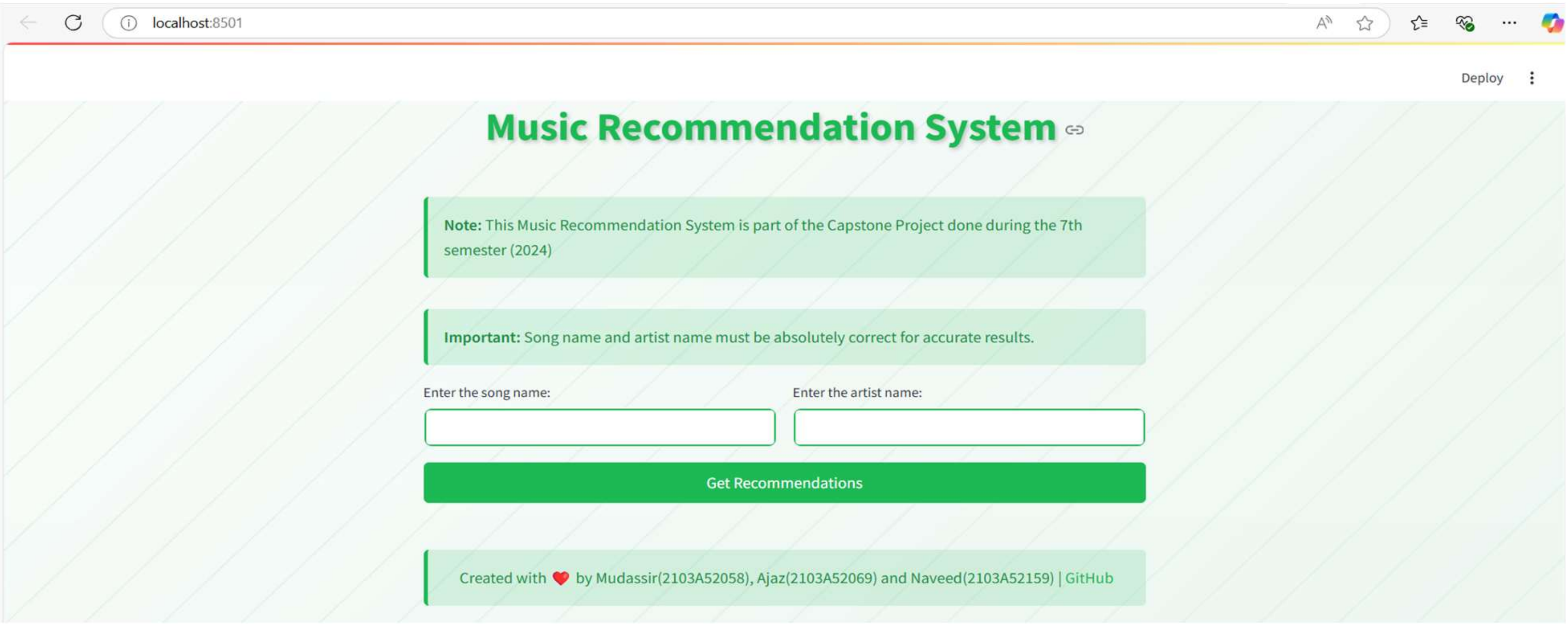

Figure 4: User Interface

RESULTS:

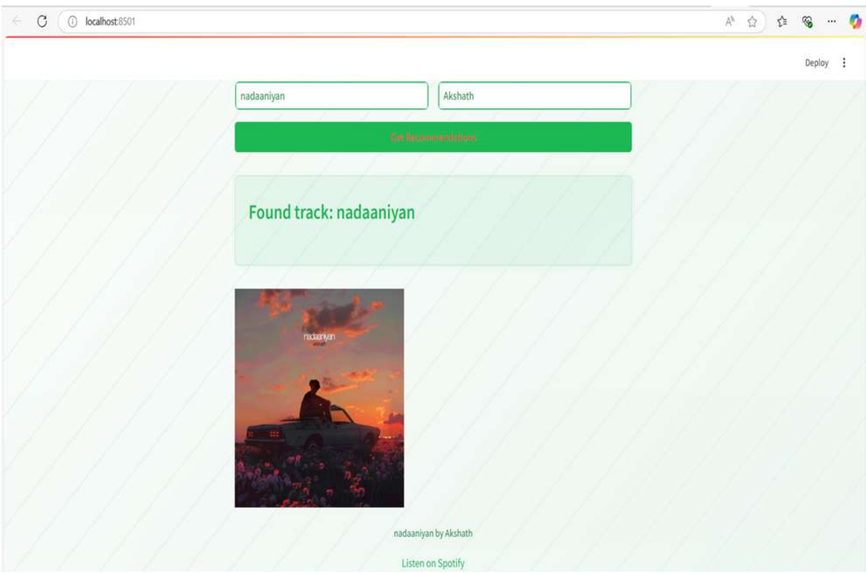

Figure 5: After giving input(track and artist name)

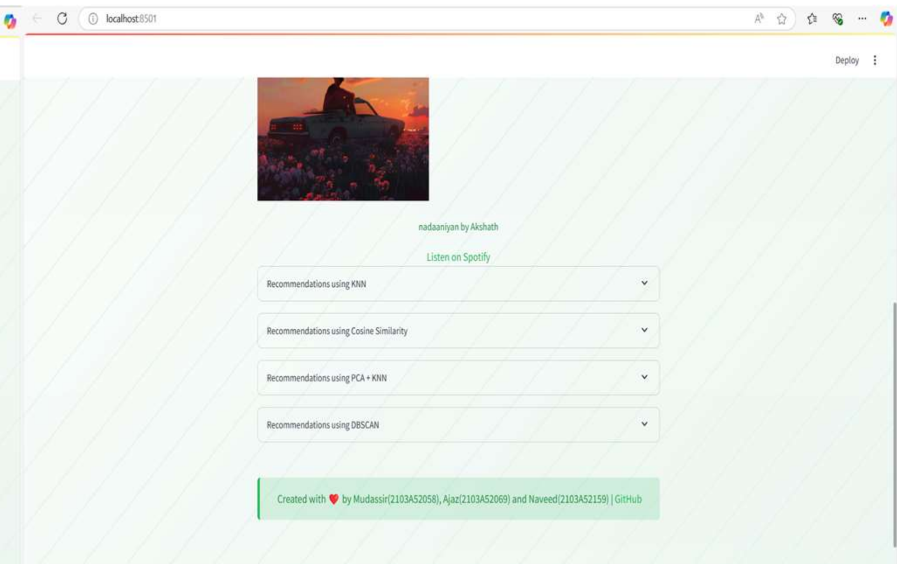

Figure 6: Recommendations are generated as containers for each model

RESULTS:

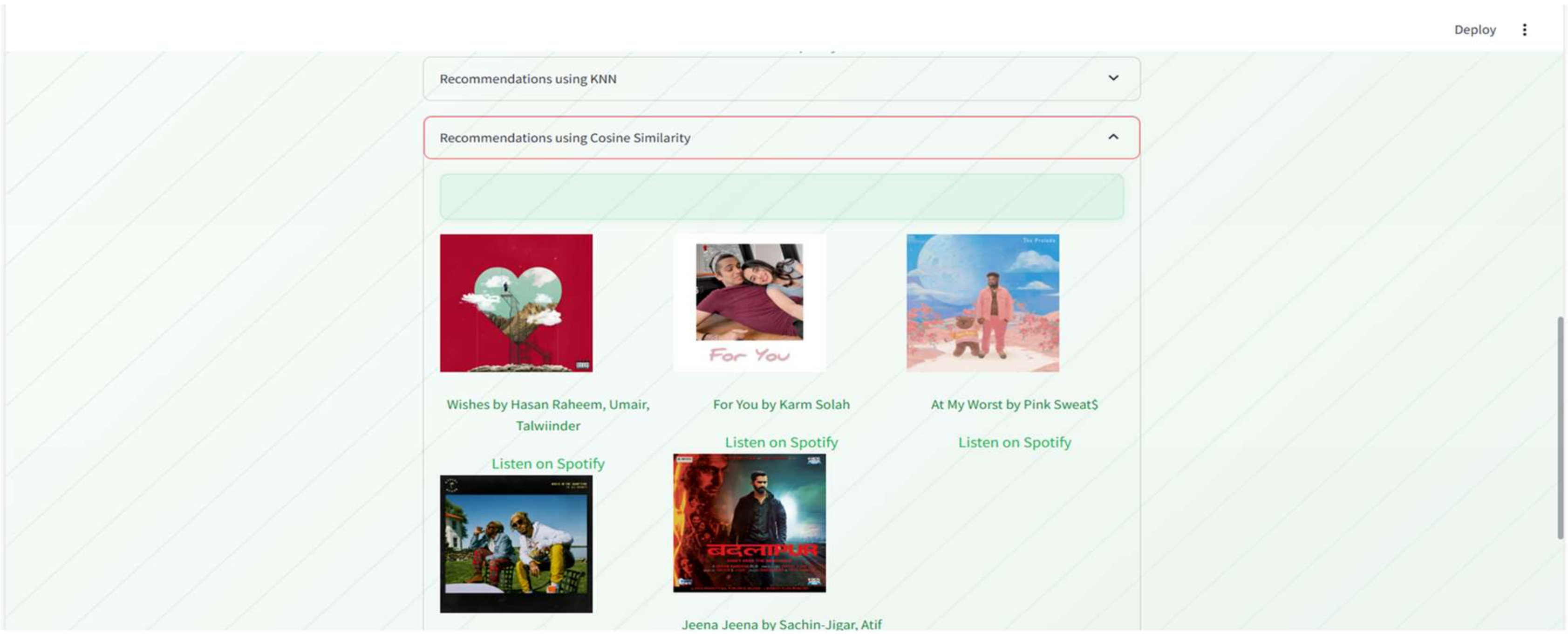

Figure 7: Recommendations given by Cosine Similarity in its container

CONCLUSION

- Successfully developed a personalized music recommendation system leveraging Spotify API and Kaggle datasets.
- Implemented four models—KNN, Cosine Similarity, PCA with KNN, and DBSCAN—offering diverse and scalable recommendation approaches.
- Addressed challenges like the cold-start problem, scalability, and real-time updates, ensuring accurate and adaptive recommendations.
- Streamlit-based user interface enabled seamless interaction and dynamic data integration.
- The project provides a robust foundation for scalable and adaptive music recommendation systems with real-time capabilities.

FUTURE SCOPE

•Incorporate user interaction data like listening history and ratings for personalized recommendations.

•Combine collaborative filtering with content-based methods for hybrid recommendation models.

•Enable real-time personalization with feedback mechanisms such as thumbs up/down.

•Expand data sources by integrating platforms like YouTube Music and SoundCloud.

•Use advanced clustering algorithms like K-Means for better song groupings.

•Leverage deep learning for mood and emotion detection in audio analysis.

•Develop a mobile application for greater accessibility and user engagement.

References:

[1] Girsang, A. S., & Wibowo, A. (2021, February). Neural collaborative for music recommendation system. In IOP Conference Series: Materials Science and Engineering (Vol. 1071, No. 1, p. 012021). IOP Publishing.

[2] Kostrzewa, D., Chrobak, J., & Brzeski, R. (2024). Attributes Relevance in Content-Based Music Recommendation System. Applied Sciences, 14(2), 855.

[3] Deldjoo, Y., Schedl, M., & Knees, P. (2024). Content-driven music recommendation: Evolution, state of the art, and challenges. Computer Science Review, 51, 100618.

[4] Bi, X., Qu, A., & Shen, X. (2018). Multilayer tensor factorization with applications to recommender systems.

[5] Zhang, Y., Bi, X., Tang, N., & Qu, A. (2020). Dynamic Tensor Recommender Systems. IEEE Transactions on Knowledge and Data Engineering, 32(5), 920-934.

THANK YOU

GitHub Link: https://github.com/Naveed-4/Music-Recommendation-System