ISSN: 2349-6002

A Data Science Approach to Personalized Music Recommendation on Spotify Using Cosine Similarity And K-Means Clustering

Tanu Jaiswal[1], Harmandeep Kaur[2], Harmanpreet Kaur[3], Pratha Saxena[4]
[1,2,3,4]*Department of Computer Applications, Invertis University, Bareilly*
doi.org/10.64643/IJIRTV12I11-207266-459

***Abstract*—Today, music streaming platforms offer access to a vast number of songs, making it difficult for users to find songs that match their interests. In this project, we built a music recommendation system that can suggest songs according to user preferences. We used a dataset from Kaggle. It includes features such as danceability, energy, tempo, and valence. These features help analyse songs and better understand their characteristics. To find similar songs, cosine similarity was applied. On the other hand, K-Means clustering helped group songs. This makes it easier to understand user preferences in a better way. The combination of similarity filtering and clustering improves the recommendations and makes them more useful by providing accurate song suggestions Different graphs and visualisations are also used to understand the dataset and feature relationships. The results show that combining similarity and clustering gives better recommendations. This system can be useful in real-world music applications and improves the overall user experience. This study shows how machine learning and clustering methods can be effectively applied to develop scalable song recommendation systems.**

***Index Terms*—Cosine Similarity, Data Science, Data Visualisation, Feature Analysis, K-Means Clustering, Music Recommendation System, Machine Learning, Spotify Dataset**

I. INTRODUCTION

In the modern digital era, web applications play a significant role in everyday life, especially in areas such as entertainment, e-commerce, and social media. Among these, music streaming platforms have gained immense popularity due to their ability to provide users with personalised experiences. However, with the exponential growth of available music content, users often face difficulty in discovering songs that match their preferences. *This challenge has led to the development of intelligent recommendation systems that utilise data science and machine learning techniques [13], [14].*

Traditional recommendation approaches rely on basic filtering methods, which are often insufficient to capture complex user behaviour and music characteristics. As data science has improved over time, it is now possible to analyse large-scale datasets containing multiple features such as danceability, energy, tempo, and user interaction patterns. These features help the system understand how different songs relate to user preferences in a more meaningful way [12].

Machine Learning plays an important role in building such intelligent systems. This helps the system better understand user preferences. Techniques such as cosine similarity are used to compare songs based on their features, thereby supporting content-based recommendations [1]. Clustering techniques like K-Means are also useful, as they group similar songs together. This makes it easier for the system to suggest songs from the same group, improving both variety and personalisation [11].

Data preprocessing is also an important step. It includes cleaning the data, selecting important features, and scaling values, which are necessary to improve performance and accuracy of the model. Visualisation tools such a pie charts, bar graphs, and heatmaps also help in understanding patterns and relationships in the data [6].

This project is about building a web-based music recommendation system using data science techniques. This system considers both behaviour and song features to provide useful recommendations. By

combining different machine learning methods, the system aims to give better performance and smoother user experience.

II. OBJECTIVES

The primary objective of this research is to develop an efficient and personalized music recommendation system using data science and machine learning techniques. The system aims to analyze song features and user behavior to provide accurate and relevant song suggestions.

The specific objectives of this study are as follows:

- To analyze and preprocess music datasets for effective data handling and feature extraction.
- To utilize cosine similarity for identifying similarity between songs based on audio features.
- To visualize data using graphs and plots for better understanding of feature distribution and relationships.
- To apply K-Means clustering to group similar songs and identify patterns within the dataset.To enhance recommendation accuracy by combining similarity-based and clustering-based approaches.
- To design a scalable and efficient system that improves user experience in music recommendation platforms.

III. LITERATURE REVIEW

Recommendation systems have become an essential component of modern web applications, especially in domains such as music streaming and entertainment. Various machine learning techniques have been proposed to improve the accuracy and personalization of recommendations [13], [14].

Earlier approaches mainly relied on collaborative filtering, which analyzes user behavior such as ratings and interactions. However, these methods often suffer from issues like cold start and data sparsity. To overcome these limitations, content-based filtering techniques were introduced, which utilize item features instead of user history [14].

Several studies have highlighted the effectiveness of cosine similarity in recommendation systems. It is widely used to measure similarity between items based on feature vectors and has shown good performance in music and movie recommendation tasks [1]. This approach enables the system to recommend items that share similar characteristics.

Clustering techniques, particularly K-Means clustering, have also been widely applied in recommendation systems. K-Means helps in grouping similar items into clusters, which improves scalability and allows efficient recommendation from large datasets [11]. It also helps in identifying patterns within the data.

Recent research focuses on hybrid recommendation systems that combine multiple techniques such as similarity-based filtering and clustering. These hybrid models provide better accuracy and diversity compared to single-method approaches [3], [4].

Machine learning techniques have also been widely applied in various domains such as agriculture and prediction systems, demonstrating their effectiveness in handling large datasets and improving decision-making processes [5], [8]. Additionally, recent studies have explored the integration of Internet of Things (IoT) with data-driven applications, further enhancing system efficiency and real-time data analysis [7]. Advanced models such as word embedding techniques have also been used in recommendation systems to improve similarity detection and recommendation accuracy [15].

Based on these studies, this research integrates cosine similarity and K-Means clustering to develop an efficient and personalized music recommendation system.

IV. METHODOLOGY

The proposed music recommendation system is developed using a structured approach involving data preprocessing, feature selection, similarity computation, and clustering techniques. The dataset used in this study was obtained from a Kaggle project by Orestas Dulinskas [10].

A. Data Collection

The first step in the system involves collecting a dataset containing various song attributes. The dataset includes multiple CSV files such as Spotify features data, tracks data, artists data, and albums data. These datasets contain important information like danceability, energy, loudness, tempo, genre, and popularity [10].

B. Data Preprocessing

After data collection, preprocessing is performed to ensure data quality and consistency. This step includes handling missing values, removing duplicate records, and converting data into a structured format.

Proper preprocessing is essential to improve the performance of machine learning models [6].

C. Feature Selection

In this stage, relevant features are selected from the dataset for building the recommendation model. Important features include danceability, energy, loudness, speechiness, acousticness, instrumentalness, valence, tempo, and time signature. These features help in identifying similarities between songs.

D. Feature Scaling

Since the selected features have different ranges, feature scaling is applied using techniques such as Standard Scaler. This ensures that all features contribute equally to the model and prevents bias due to varying scales [1].

E. Similarity Calculation (Cosine Similarity)

Cosine similarity is used to measure the similarity between songs based on their feature vectors.

It calculates the cosine angle between two vectors and provides a similarity score. Songs with higher similarity scores are considered more relevant for recommendation [1].

F. Clustering using K-Means

K-Means clustering is applied to group similar songs into clusters based on their features. The optimal number of clusters is determined using the Elbow Method.

Songs belonging to the same cluster share similar characteristics, which helps in improving recommendation diversity [11].

G. Recommendation Generation

The system generates recommendations using both cosine similarity and clustering techniques. For a given input song, similar songs are identified based on similarity scores and cluster grouping.

The top similar songs are then recommended to the user.

H. Visualization and Analysis

Visualization techniques such as pie charts, bar graphs, and heatmaps are used to analyze the dataset and understand feature relationships. These visualizations help in validating the model and improving system performance.

I. Web Application Integration

Finally, the recommendation system is integrated into a web-based application where users can input a song name and receive personalized recommendations. The system ensures efficient performance and user-friendly interaction

V. RESULTS AND DATA VISUALIZATION

This section presents the analysis and visualization of the dataset used for building the music recommendation system. Various graphical techniques are used to understand patterns, feature distributions, and clustering results.

A. Content Analysis

The Fig.1 consists of both explicit and non-explicit songs. The distribution analysis reveals that a majority of songs are non-explicit, indicating that most of the music content is suitable for a general audience.

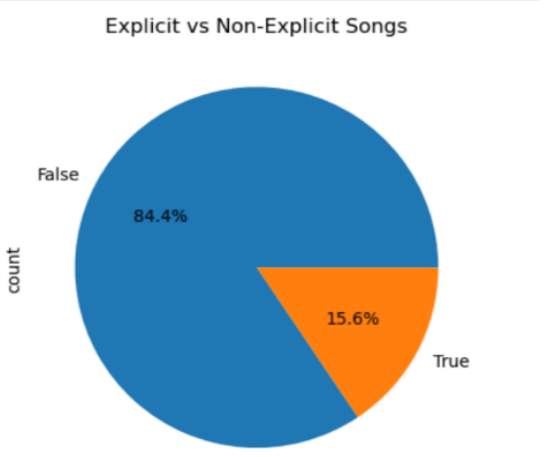

Fig.1: Distribution of Explicit and Non-Explicit Songs

B. Musical Feature Analysis

The Fig.2 shows that most songs belong to the major mode, which is generally associated with positive and energetic emotions. This highlights the overall nature of songs present in the dataset.

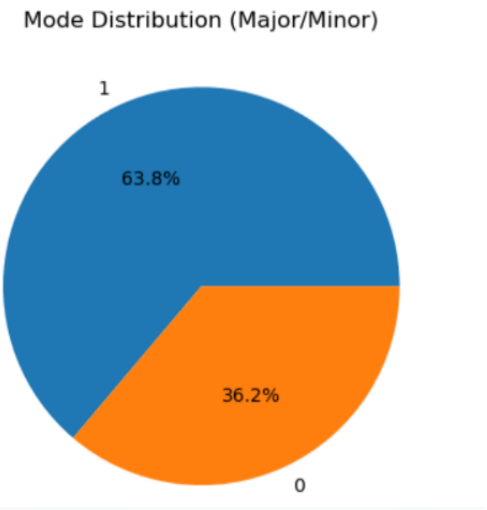

Fig.2: Distribution of Song Modes (Major/Minor)

C. Time Signature Distribution

The Fig.3 illustrates the distribution of time signatures in the dataset. It is evident that the majority of songs have a time signature of 4, which is commonly used in modern music.

Other time signatures are present in smaller proportions.

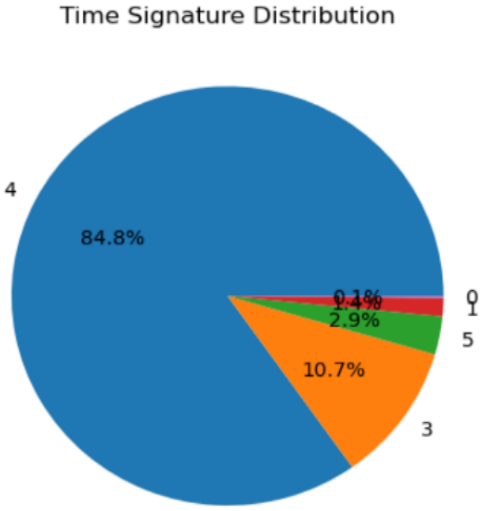

Fig.3: Time Signature Distribution

D. Distribution of Musical Keys

The Fig.4 represents the distribution of musical keys across songs. The dataset contains a variety of keys, indicating diversity in musical composition.

Some keys are more dominant, showing commonly used tonal structure.

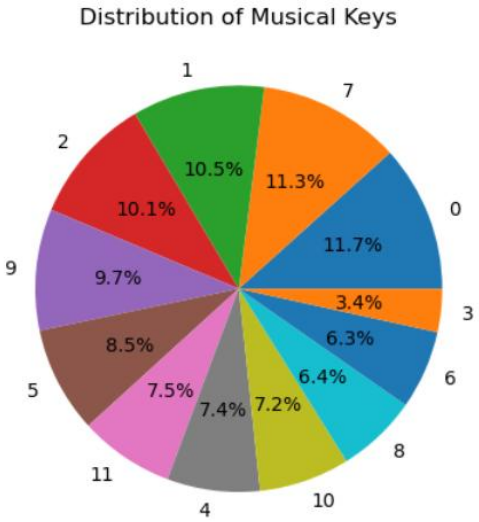

Fig.4: Musical Keys Distribution

E. Distribution of Danceability

The Fig.5 shows distribution of danceability in the graph. It can be observed that most songs have danceability values between 0.4 and 0.8, indicating that a large portion of songs are moderately danceable. This suggests that users generally prefer songs with balanced rhythm and tempo suitable for dancing.

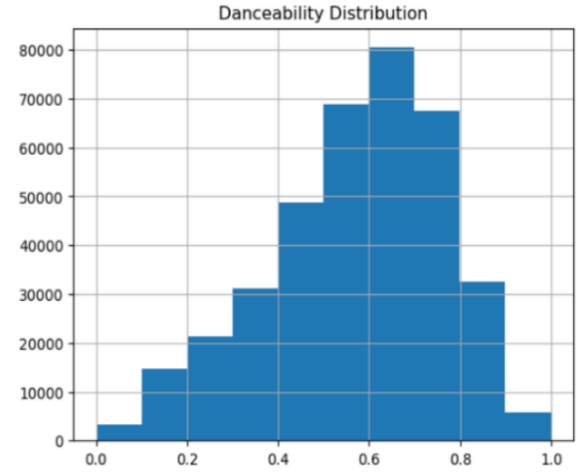

Fig.5: Distribution of Danceability

F. Density Distribution of Danceability

The Fig.6 shows the density plot of danceability provides a smooth representation of its distribution. The curve shows a peak in the mid-range values, confirming that the dataset follows a near-normal distribution.

This helps in understanding the central tendency of danceability among songs.

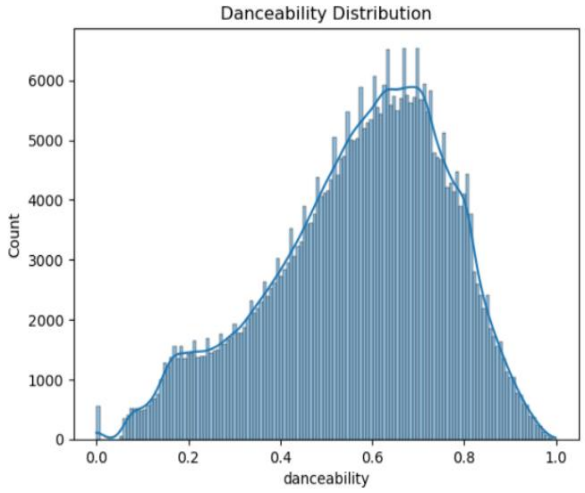

Fig.6: Danceability Distribution

G. Relationship between Energy and Danceability

The Fig.7 represents the relationship between energy and danceability. The data points are widely distributed, indicating that there is no strong linear relationship between the two features. This highlights the diversity in musical compositions, where both high-energy and low-energy songs can vary in danceability.

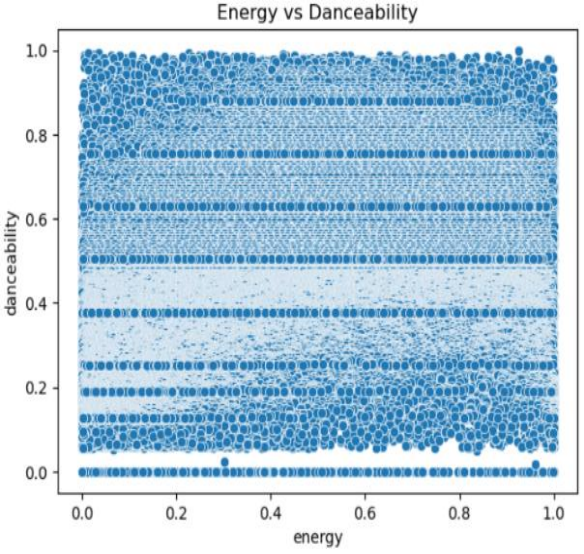

Fig.7: Energy Vs Danceability

H. Top 10 Popular Songs

The Fig.8 shows the top 10 songs based on their popularity scores. These songs have significantly higher popularity compared to others, reflecting current trends and user preferences. This analysis helps in identifying frequently streamed and highly preferred tracks.

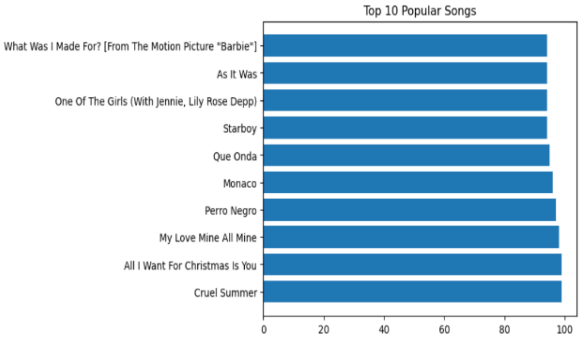

Fig.8: Top 10 Popular Songs

I. Correlation Matrix of Audio Features

The Fig.9 shows the relationships between different audio features such as danceability, energy, loudness, tempo, and acousticness. A strong positive correlation is observed between energy and loudness, while some features show weak or negative relationships. This analysis helps in understanding how features interact and supports better recommendation performance.

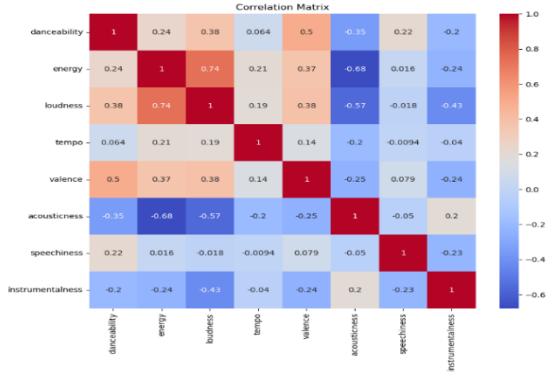

Fig.9: Correlation Matrix

J. Genre Analysis

The Fig.10 provides insights into the most popular music categories. The bar graph indicates that genres like karaoke, dance pop, and classical are more dominant in the dataset, reflecting user listening preferences.

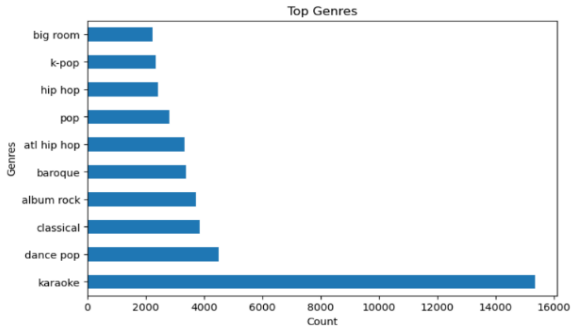

Fig.10: Top Music Genres in Dataset

K. Clustering Optimization

The Elbow Method is used to determine the optimal number of clusters for the K-Means algorithm. The Fig.11 shows a significant bend at k = 10, indicating that this value is suitable for grouping similar songs effectively.

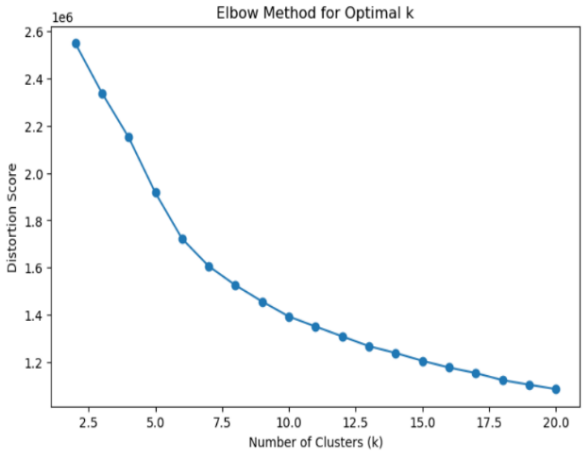

Fig.11: Elbow Method for Optimal Number of Cluster

L. Clustering Analysis

K-*Means* clustering is applied to group songs based on audio features such as danceability and energy. The Fig.12 visualizes different clusters, where each cluster represents songs with similar characteristics. This helps in improving recommendation accuracy.

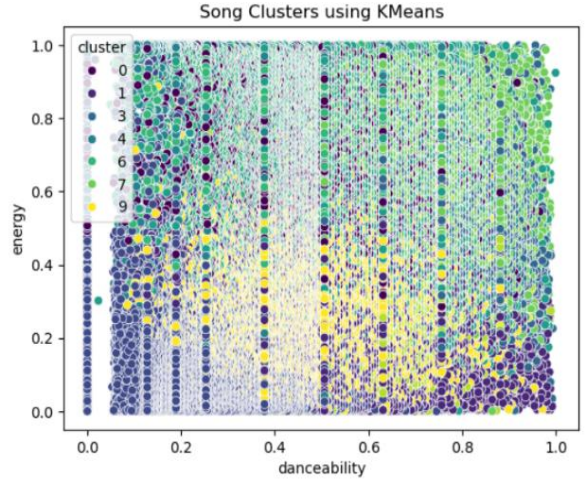

Fig.12: Song Clusters Using K-Means Algorithm

The above visualizations help in understanding the dataset structure and validating the effectiveness of the recommendation model. These insights play a crucial role in enhancing personalised music recommendations.

VI. CONCLUSION

In this research, a music recommendation system was developed using data science and machine learning techniques. The system utilizes audio features such as danceability, energy, tempo, and valence to analyze songs and generate recommendations. Cosine similarity was used to measure similarity between songs, while K-Means clustering was applied to group similar songs into clusters.

The results show that combining similarity-based and clustering techniques improves the overall recommendation performance. The use of data visualization techniques such as histograms, scatter plots, pie charts, and heatmaps helped in understanding the dataset and identifying meaningful patterns.

Overall, the proposed system is efficient, scalable, and capable of providing personalized music recommendations based on song features.

The current system can be further improved by incorporating advanced techniques and additional data sources.

Future enhancements may include:

- Integration of user listening history for better personalization
- Implementation of deep learning models for improved accuracy
- Real-time recommendation using streaming data
- Inclusion of lyrics and sentiment analysis
- Deployment as a fully functional web or mobile application

These improvements can make the recommendation system more intelligent, adaptive, and user-centric.

REFERENCES

[1] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[2] D. Pranita, "Recommendation system using machine learning techniques," in Proc. Int. Conf. Computing and Data Science, 2021.

[3] M. Hasan and J. Ferdous, "Hybrid movie recommendation system using machine learning," in Proc. IEEE Int. Conf. Advanced Computing, 2023.

[4] M. Nasri, “A hybrid approach for movie recommendation systems,” in Proc. Int. Conf. Artificial Intelligence, 2022.
[5] A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” Comput. Electron. Agric., vol. 147, pp. 70–90, 2018.
[6] K. G. Liakos et al., “Machine learning in agriculture: A review,” Sensors, vol. 18, no. 8, Art. no. 2674, 2018.
[7] X. Zhang et al., “Internet of Things for smart agriculture,” in Proc. Int. Conf. IoT Systems, 2019.
[8] A. Chlingaryan, S. Sukkarieh, and B. Whelan, “Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review,” Comput. Electron. Agric., vol. 151, pp. 61–69, 2018.
[9] Spotify Dataset, Kaggle. [Online]. Available: https://www.kaggle.com/
[10] O. Dulinskas, “Song recommendation system,” Kaggle, 2023. [Online]. Available: https://www.kaggle.com/code/orestasdulinskas/song-recommendation-system
[11] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[12] X. Amatriain, “Building industrial-scale real-world recommender systems,” in Proc. ACM Conf. Recommender Systems (RecSys), 2013.
[13] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,” IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734–749, Jun. 2005.
[14] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, “Collaborative filtering recommender systems,” in The Adaptive Web, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Berlin, Germany: Springer, 2007, pp. 291–324.
[15] T. Mikolov et al., “Efficient estimation of word representations in vector space,” in Proc. Int. Co*nf. Learn. Representations (ICLR)*, 2013.