Harmonizing Hearts: A Music-Based Compatibility Analysis

Abhinav Acharya Tirumala Vinjamuri †, Arihant Dugar ‡, Dhruv Patel ¥, Mudra Verma*

†ab806657@dal.ca (B00929073), Faculty of Computer Science, Dalhousie University.

‡ arihant.dugar@dal.ca (B00917961), Faculty of Computer Science, Dalhousie University.

¥ dhruvpatel@dal.ca (B00891974), Faculty of Computer Science, Dalhousie University.

* mudraverma@dal.ca (B00932103), Faculty of Computer Science, Dalhousie University.

GitHub Link: https://github.com/mudraverma65/harmonizing_hearts

Note: In adherence to the procedures outlined by other courses, our project initially utilized a private GitLab repository to manage version control and collaboration. Subsequently, upon reviewing the provided instructions, it became evident that a public GitHub repository was required for the project. Consequently, the transition to GitHub resulted in the absence of commit history on the public repository, as the comprehensive commit records were originally maintained within the confines of the private GitLab repository.

Abstract:

This project, "Harmonizing Hearts," explores the intersection of music and online dating, investigating whether shared musical tastes influence relationship dynamics [1]. Leveraging Spotify data, our study employs advanced algorithms for user matching based on artists or song features, revealing clusters of individuals with similar musical preferences, and discover the popular songs according to various metrics. In parallel, we employ K-means clustering to organize songs by audio features, allowing users to explore cohesive musical spaces.

The results illuminate the significance of music in online dating, showcasing the potential for shared musical experiences to foster connections. The user matching algorithm demonstrates the feasibility of identifying like-minded individuals based on common shared items such as liked artists or detailed song features, while the song clustering and song popularity unveil the diversity and commonalities within musical landscapes. Overall, "Harmonizing Hearts" offers a nuanced perspective on the role of music in online relationships, providing a foundation for understanding how musical affinity contributes to the formation and enhancement of connections in the digital dating realm. There are existing applications solely for online dating our study focuses on clustering data and making popularity classifications based on patterns observed [2].

Keywords: Online dating, Machine Learning, K-means clustering, Spotify, music, dimensionality reduction, K-nearest neighbors, random forest, genre identification, Collaborative Filtering, Content-Based Filtering, recommendation algorithms.

1. Introduction

"Harmonizing Hearts: A Music-Based Compatibility Analysis" delves into the intricate interplay between music preferences and online dating dynamics. Leveraging Spotify data, this project explores the hypothesis that shared musical tastes contribute significantly to relationship success in the online dating realm. Our study also focuses on exploring the data by clustering and making popularity classifications. The study encompasses two primary objectives.

In Part 1, "Similar User Matching," an advanced recommendation algorithm is refined through hyperparameter tuning, utilizing a comprehensive set of song features such as danceability and energy. The implementation incorporates collaborative filtering and content-based filtering to identify users with similar music preferences.

Part 2 unfolds in two segments: (2a) "Song Clustering and Exploration" and (2b) "Song Popularity". In 2a, songs are clustered based on their audio features using algorithms like K-means. Users are provided with an interactive interface to explore songs within clusters. In 2b, a multitude of models are trained, and classification is made to determine the popularity of songs. Popularity score of more than 70 indicate popularity in songs.

Through this multifaceted approach, our project endeavours to unravel the nuanced role of music in forging connections within the realm of online dating, shedding light on whether a shared melody can truly harmonize hearts. It aims to study patterns in song features and perform explorations. Finally, our approach is also focused on finding patterns and how they relate to making songs popular. We wish to train multiple models and perform classifications in determining the popularity of songs based on presence of certain features [3].

* corresponding_author@example.ca

Datasets Used:

1. Dataset 1: spotify_data.csv (Large Dataset) [4]
- This is a dataset with 250 anonymous Spotify users, each with around 100 songs. The total number of rows in this dataset is 28469. Each song's features such as danceability, valence, key, energy, etc. are extracted from Spotify API. This dataset was created for applying recommender models using collaborative and content-based filtering. The goal was to match similar users for a user using collaborative and content-based filtering models.

- Dataset Source: Spotify Dataset (kaggle.com).

- Dataset inspired and extended from https://www.kaggle.com/datasets/andrewmvd/spotify-playlists?select=spotify_dataset.csv.

2. Dataset 2: track_features.csv (Large Dataset) [6]
- This Dataset consists of 17,996 rows, each representing a unique song, with 17 columns detailing various attributes of the songs. These attributes include the artist's name, track name, popularity, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, duration in milliseconds, and time signature. The target variable is 'Class', which categorizes each song into one of several genres such as Rock, Indie, Alt, Pop, Metal, HipHop, Alt_Music, Blues, Acoustic/Folk, Instrumental, Country, and Bollywood. The test dataset, used to evaluate the model's performance, contains 7,713 rows with the same 16 feature columns as the training set.

- Dataset Source: https://www.kaggle.com/datasets/rodolfofigueroa/spotify-12m-songs.

3. Dataset 3: spotifydata.csv (Large Dataset) [5]
- T his Dataset analyzes the behavior between valence and all the measures that Spotify API gives so I got approximately 10,000 per genre. There are 26 genres, so it is a total of 232,725 tracks. This dataset has key, mode, and time signature attributes that are cleaned. In total, it has 18 columns, out of which, we will use some of the most prominent features to predict the popularity of the song based on a threshold we are setting.

- Dataset Source: https://www.kaggle.com/datasets/zaheenhamidani/ultimate-spotify-tracks-db.

2. Literature Review

A. Similar User Matching

We are in times where the realm of online dating is seen as the go to solution for finding partners. The traditional approach of finding potential partners with respect to hobbies and skills has been replaced by songs, series, and books. Delving further and finding partners who appreciate the same art not just the art form. The paper by Edward B Kang [2] adopts a cultural anthropological perspective to investigate the intersection of music taste and self-presentation on Tinder through its integration with Spotify. It addresses the dearth of literature centered on music taste as a communicative tool in online dating, shedding light on the increasingly influential role of music on emerging dating platforms.

B. Song clustering and exploration

The study by Joshua S, Julie S, Mon M, and others employs three models—K-means, Linear Regression, and Random Forest—to evaluate auditory aspects of songs, showcasing a commitment to diverse analytical methods in music classification [3]. The Linear Regression model, with its low mean error rates during training and testing, and the Random Forest Model's impressive 95.37 percent accuracy underscore the study's dedication to robust modeling techniques. The paper suggests future improvements through the inclusion of additional metadata, hinting at the potential for enhanced accuracy and outcomes in music analysis.

C. Song Popularity

The existing study focuses on classifying song popularity for the Indonesian market [4]. It determined the impact of features such as energy, speechiness, acousticness, danceability, liveness, and time signature's impact on popular songs. It mapped features that contributed to popularity as per genres and identified patterns. Furthermore, the study trained a random forest model and categorized the songs into either "popular" or "unpopular". The model's accuracy was 72.8% with five significant features for the classification i.e. acousticness, liveness, energy, valence, and key.

3. Methodology

A. Similar User Matching

The Spotify dataset, containing key audio features of songs, forms the cornerstone of our analysis. With features such as danceability, energy, and tempo, the dataset encompasses 28469 entries. To streamline the dataset, irrelevant columns were removed from the Spotify dataset. Furthermore, textual columns were encoded into numerical representations, facilitating subsequent model training.

A suite of visualizations, including box plots, histograms, and a correlation heatmap, offered in-depth insights. Box plots exposed potential outliers, histograms detailed feature distributions, and the heatmap illuminated correlations between numerical features.

The decision to not handle or transform the abnormalities in the dataset, such as outliers and correlations, is a valid approach in certain scenarios, especially in the context of recommendation systems. Here's a more detailed explanation:
1. **Preserving Originality:** Outliers in a dataset often carry significant information. In the context of songs, these outliers could represent unique characteristics that set a song apart. Handling these outliers could distort these unique features, thereby affecting the originality of the song.
2. **Maintaining Information Integrity:** High correlation between columns in a dataset usually indicates redundancy. However, in the case of recommendation systems, even highly correlated features can provide valuable insights. Removing these columns could lead to loss of crucial information, which could negatively impact the quality of the recommendations.
3. **Importance of Abnormalities:** The so-called "abnormalities" in the dataset might represent important patterns or trends. For instance, a group of songs that are outliers in terms of a certain feature might be a specific genre or style of music. Similarly, correlated features might indicate a common trend in music preferences.
4. **Context-Specific Decisions:** The decision to handle or ignore these abnormalities is highly dependent on the specific use case. In machine learning, there's no one-size-fits-all approach. The best strategy is often determined by the nature of the data and the specific requirements of the project.
5. **Visualizations for Insight, Not Action:** Visualizations are a powerful tool for understanding the structure and patterns in the data. However, not all observations from visualizations necessitate action. In this case, the visualizations serve to highlight the characteristics of the dataset, but do not necessarily imply a need for data transformation or cleaning.

In conclusion, the choice to preserve the dataset in its original form, despite the observed abnormalities, is a strategic decision that prioritizes information preservation and the integrity of the recommendation system. This approach is indeed common in the field of machine learning, particularly in scenarios where the cost of information loss outweighs the potential benefits of data cleaning or transformation.

Normalization played a crucial role, ensuring numerical features were uniformly scaled. For model training two methods for filtering were deployed. The detailed descriptions for both are as follows:

a. Collaborative Filtering -
   We began by creating a user-artist matrix, encoding user interactions. A collaborative filtering approach involved computing user similarity using the Jaccard similarity metric. Relevant recommendations were generated by setting a threshold for the minimum number of common artists between users.

b. Content-Based Filtering –
   Utilizing audio features, a content-based filtering approach assessed user similarity through cosine similarity. Relevant recommendations were derived by comparing user-specific features and applying a similarity threshold.

Collaborative filtering demonstrated precision of 1.0, indicating accurate recommendations, albeit with a modest recall. Content-based filtering achieved balanced precision and recall, with an F1 score of 0.71. These findings suggest collaborative filtering's strength in precision, while content-based filtering strikes a balance between precision and recall.
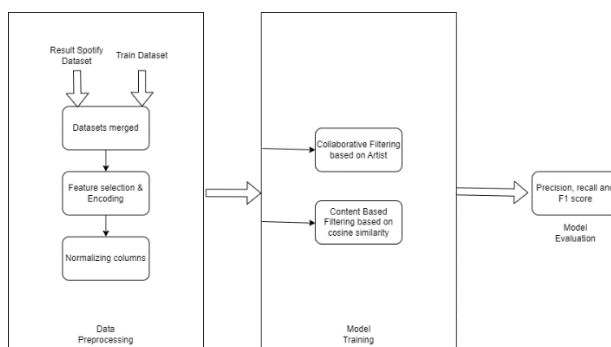

Figure 1. Similar User Matching Architecture.

B. Song clustering and Exploration

We initiated our analysis by loading the 'tracks_features.csv' dataset and conducting an EDA. The dataset, containing features of various tracks, was examined for missing values. We decided to drop rows with missing data and then sampled 25,000 entries for further analysis. Key features, including 'explicit,' 'danceability,' 'energy,' 'loudness,' and others, were selected for clustering analysis. To ensure uniformity, we normalized the values of each selected feature. This step is crucial in preventing features with larger scales from dominating the clustering process.

The K-Means clustering algorithm was employed to group tracks into 8 clusters based on the selected features. The Elbow Method, visualizing the sum of squared distances for different cluster numbers, guided our choice of 8 clusters. The algorithm was then fitted to the data, and clusters were assigned to each track. To assess the quality of our clustering solution, we calculated the silhouette score. A score of approximately 0.14 was obtained, suggesting that the clusters have moderate cohesion and separation. This indicates that while some patterns exist, the clusters might not be well-defined, potentially due to the nature of the data.

Figure 2. Song clustering and Exploration Architecture.

C. Song Popularity

The 'spotifydata.csv' dataset underwent comprehensive exploratory data analysis (EDA). The analysis covered data overview, distribution of popularity, correlation insights, and the impact of features like time signature, key, mode, acousticness, and loudness on song popularity. Next, the 'key' and 'time_signature' columns were numerically encoded to simplify their representation. The 'mode' column was transformed into binary encoding, replacing 'Major' with 1 and 'Minor' with 0. The 'popularity' column was binarized, categorizing songs with a popularity score below 70 as 0 and those equal to or above 70 as 1. Finally, a list named 'song_features' was created to encompass relevant song attributes, laying the groundwork for subsequent analysis and modeling with a focus on predictive tasks.

A predictive modeling approach was applied to predict the popularity of songs based on selected features. The dataset was split into training and validation sets, and various classification algorithms were employed, with performance metrics evaluated.
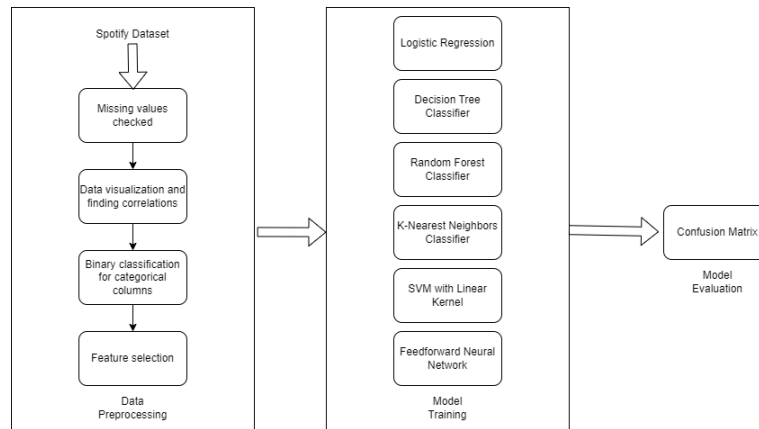

Figure 3. Song popularity Architecture.

4. **Results**

A. Similar User Matching
   a. Collaborative Filtering Evaluation
      For collaborative filtering with user_id 100, the analysis identified the topmost similar users based on percentage similarity. The results are as follows:
      - Similar User 14: 13.21% similarity
      - Similar User 11: 9.17% similarity
   b. Content-Based Filtering Evaluation

In content-based filtering, the top 5 similar users for the given user_id were identified based on similarity percentage. The results are as follows:

- Similar User 1: 100.0% similarity
- Similar User 95: 100.0% similarity

Model	Precision	Recall	F1 Score
Collaborative Filtering	1.0	0.13	0.24
Content Based Filtering	0.85	1.0	0.92

Table 1. Model Comparison Results.

As shown in figure 2 the recommendations made for collaborative filtering were relevant but only a fraction of the relevant items was successfully recommended. Whereas, in content-based filtering the perfect scores indicate that all the recommendations were not only relevant but also successfully identified all the relevant items. This implies that the content-based filtering algorithm is highly effective for the given user_id, achieving better precision, recall, and F1 Score.

B. Song clustering and Exploration

The clustering process seeks to group tracks with similar features, aiding in music categorization into 8 clusters. However, a silhouette score of 0.14 implies that certain tracks may not fit neatly into distinct clusters. This ambiguity may arise from the inherent complexity of music genres, where songs often exhibit characteristics of multiple genres, challenging precise categorization. The exploration provides valuable insights into the potential clustering patterns of tracks based on audio features.
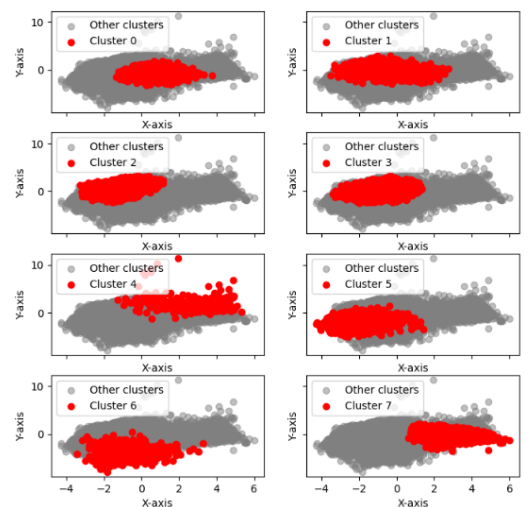

Figure 4. K-means clustering into 8 clusters.

C. Song popularity

The evaluation of the models trained are as follows:

Model	Accuracy	AUC
Logistic Regression	96.01%	50%
Decision Tree	96.75%	84.42%
Random Forest	98.62%	84.33%
K-Nearest Neighbors	95.45%	56.99%
SVM with linear kernel	91.85%	56.38%

Neural Network	95.99%	78.02%

Table 2. Model Comparison Results

These results highlight the varying performance of different algorithms in predicting song popularity. The Random Forest model exhibited the highest accuracy, while the Decision Tree and Neural Network models demonstrated good discriminatory power. Consideration of these findings is crucial for selecting an appropriate model based on the research objectives and desired trade-offs between accuracy and interpretability.

5. Discussion

A. Similar User Matching
   a. Collaborative Filtering -
      The collaborative filtering approach successfully identified users with similar song preferences to the target user (user_id 100). The precision of 1.0 indicates that all recommended users were indeed relevant, demonstrating the accuracy of the recommendations. However, the relatively low recall of 0.133 suggests that while the system is precise, it may miss a considerable portion of users with similar preferences. The F1 score of 0.235, a harmonic mean of precision and recall, highlights the trade-off between precision and recall in the collaborative filtering model. This emphasizes the need for careful consideration of the threshold parameter, as it significantly influences the balance between precision and recall.
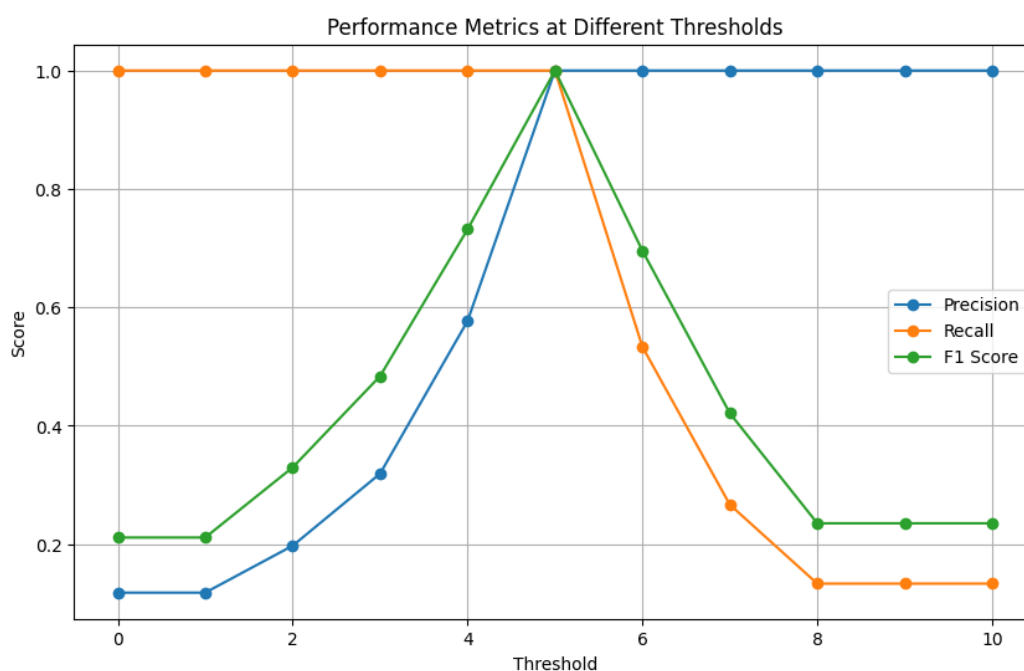

Figure 2 Evaluation metrics at different values of threshold – Collaborative Filtering

In the graph above for collaborative filtering, we are plotting the precision, recall, and F1 score at different threshold levels. These metrics are commonly used to evaluate the performance of a recommendation system:

- Precision measures the proportion of recommended items that are relevant.
- Recall measures the proportion of relevant items that are recommended.

- F1 Score is the harmonic mean of precision and recall, providing a single metric that balances both values.

Apart from that, here's a detailed analysis of the collaborative filtering recommendation model:

- Threshold Impact: The graph shows how the performance metrics (precision, recall, and F1 score) change with different thresholds. The threshold here refers to the number of artists that both users need to have in common to consider the match relevant.
- Precision and Recall Trade-off: As the threshold increases, both f1_score and recall decrease. This suggests that as the threshold increases, the system becomes more conservative and precise in its recommendations, but at the cost of recall and F1 score. This is a common trade-off in recommendation systems. A higher threshold means the system is stricter about what it considers a relevant recommendation, which can increase precision but decrease recall. The F1 score, which balances precision and recall, also decreases as the system becomes too conservative.
- Optimal Threshold: From the graph, it seems that the F1 score is highest at a threshold of 5. This suggests that a threshold of 5 results in the best balance between precision and recall, making it the most effective threshold for finding similar users in the collaborative filtering algorithm.

b. Content-Based Filtering –

Conversely, the content-based filtering method exhibited a perfect precision, recall, and F1 score, attaining 0.85, 1.0, and 0.92. This implies that the content-based model flawlessly identified and recommended users with identical song preferences. The content-based approach excels in scenarios where user preferences align strongly with specific features used in the recommendation algorithm. In this case, the chosen features, their extraction, and their interpretation played a pivotal role in achieving optimal performance.
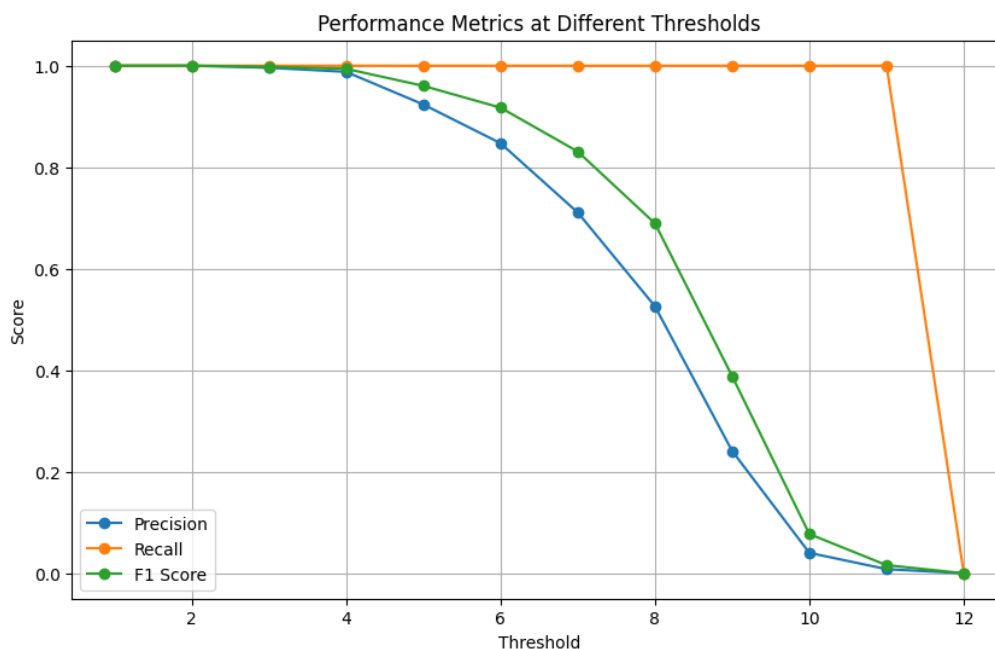

Figure 3 Evaluation metrics at different values of threshold – Content-Based Filtering

In the graph above for content-based filtering, we are plotting the precision, recall, and F1 score at different threshold levels. These metrics are commonly used to evaluate the performance of a recommendation system:

- Precision measures the proportion of recommended items that are relevant.
- Recall measures the proportion of relevant items that are recommended.
- F1 Score is the harmonic mean of precision and recall, providing a single metric that balances both values.

Apart from that, here's a detailed analysis of the content-based filtering recommendation model:

- Threshold Impact: The graph shows how the performance metrics (precision, recall, and F1 score) change with different thresholds. The threshold here refers to the number of features that both users need to have in common to consider the match relevant.
- Precision and F1_score Trade-off: As the threshold increases, both f1_score and precision decrease. This suggests that as the threshold increases, the system becomes more conservative and less precise in its recommendations, but at the cost of precision and F1 score. This is a common trade-off in recommendation systems. A higher threshold means the system is stricter about what it considers a relevant recommendation, which can decrease precision. The F1 score, which balances precision and recall, also decreases as the system becomes too conservative.
- Optimal Threshold: From the graph, it seems that the F1 score is highest at a threshold of 3. This suggests that a threshold of 3 results in the best balance between precision and recall, making it the most effective threshold for finding similar users in the content-based filtering algorithm.

Comparing the two methods, it is evident that collaborative filtering may sacrifice recall for higher precision, potentially resulting in a more conservative recommendation system. Content-based filtering, on the other hand, relies on the explicit features of songs and users, leading to a more deterministic matching process. While the results showcase promising outcomes, it is crucial to acknowledge certain limitations. Collaborative filtering's dependence on user interactions and the sparsity of data can impact performance, particularly for new users. Content-based filtering success relies heavily on the relevance of chosen features, which may not capture all nuances of user preferences. Moreover, a significant consideration must be given to the similarity function employed for either of the recommendation models. Collaborative filtering typically uses Jaccard Similarity, which measures the similarity between users based on the items they have in common. This can be more conservative and stricter in assigning a similarity score between users. In contrast, content-based filtering employs Cosine Similarity, which calculates the cosine of the angle between feature vectors. This measure is more lenient and does not prioritize any feature over others, potentially leading to more diverse recommendations. A hybrid approach that leverages the strengths of both methods could potentially mitigate these limitations and provide a more robust recommendation system.

B. Song clustering and Exploration:
In the plot, it is essential to identify the point at which adding more clusters provides diminishing returns in terms of reducing intra-cluster distances. The subsequent part of the code utilizes PCA (Principal Component Analysis) to reduce the data to two dimensions and visualizes the clusters on a scatter plot. The effectiveness of clustering relies on the choice of features and the assumption that similar audio features lead to similar music characteristics.

It may not capture all nuances of user preferences. The algorithm's sensitivity to initialization and potential convergence to local optima should be considered.

C. Song Popularity:

The selection of an appropriate model depends on the research objectives and the importance of accuracy versus interpretability. While the Random Forest model demonstrates the highest accuracy, decision-makers should weigh the interpretability of complex models against their predictive power. The Random Forest model, with its exceptional accuracy and AUC, appears to generalize well to new data. Generalization is a critical aspect, especially in applications where the model needs to perform accurately on unseen instances. The Decision Tree model also emerges as a strong contender, striking a balance between accuracy and interpretability.

Depending on the specific requirements of the application, a simpler model like Decision Tree might be preferred. Models with lower AUC, such as Logistic Regression and SVM, may benefit from feature engineering or parameter tuning to enhance their discriminatory capabilities. The evaluation should consider potential class imbalances in the dataset, as accuracy alone might be misleading. Techniques like oversampling or adjusting class weights can be explored to address this.

6. Conclusion

A. Similar User Matching:

In summary, our exploration of similar user matching using collaborative and content-based filtering revealed distinct strengths and trade-offs. Collaborative filtering excelled in precision, offering accurate recommendations, albeit with a compromise in recall. On the other hand, content-based filtering demonstrated perfection across precision, recall, and F1 Score, emphasizing its effectiveness for the given user_id. The study suggests a nuanced approach, considering hybrid models that balance the strengths of both techniques, and underscores the importance of tailoring recommendation systems to individual user preferences. As we navigate this dynamic landscape, future research should focus on refining models through user feedback integration and exploring hybrid approaches for more effective and personalized music discovery experiences.

B. Song clustering and Exploration:

In conclusion, our exploration of song clustering using KMeans on key audio features revealed moderate cohesion and separation among the identified clusters. The Elbow Method guided our decision to opt for 8 clusters, providing a structured approach to music categorization. However, the obtained silhouette score of approximately 0.14 indicates that while discernible patterns exist, the clusters lack well-defined boundaries. This ambiguity may be attributed to the inherent complexity of music genres, where songs often transcend traditional genre boundaries. Despite these challenges, our clustering analysis provides valuable insights into potential patterns within the diverse landscape of audio features, paving the way for further investigations into nuanced music categorization approaches.

C. Song Popularity:

The diverse set of classification algorithms employed revealed nuanced insights into their predictive capabilities. Notably, the Random Forest model emerged as the frontrunner, boasting the highest accuracy, while the Decision Tree and Neural Network models showcased commendable discriminatory power. The presented model comparison results underscore the importance of carefully selecting a model aligned with specific research objectives and the desired balance between accuracy and interpretability. These findings

contribute to the ongoing discourse on leveraging machine learning for music popularity prediction, offering valuable guidance for future endeavors in this domain.

References

[1] "Music taste has become the latest weapon in our online dating war | Elle Hunt," the Guardian, Jan. 28, 2022. https://www.theguardian.com/music/2022/jan/28/personalised-playlists-online-dating-west-elm-caleb

[2] R. Hager, "Try the Spotify-based dating app I am apparently too snobby for," *Android Police*, Apr. 02, 2021. https://www.androidpolice.com/2021/04/02/try-the-spotify-based-dating-app-i-am-apparently-too-snobby-for/ (Accessed: Dec. 07, 2023).

[3] V. Pareek, "Song Popularity Predictor," *Medium*, Jul. 10, 2019. [Online], Available: https://medium.com/@pareek.varuchi/song-popularity-predictor-4e784d28a368 (Accessed: Dec. 07, 2023).

[4] "Spotify Playlists," *Kaggle*. [Online], Available: https://www.kaggle.com/datasets/andrewmvd/spotify-playlists?select=spotify_dataset.csv (Accessed: Dec. 07, 2023).

[5] "Music Genre Classification," *Kaggle*. https://www.kaggle.com/datasets/purumalgi/music-genre-classification/data?select=train.csv (Accessed: Dec. 07, 2023).

[6] "Spotify 1.2M+ Songs," *Kaggle. [Online]*. Available: https://www.kaggle.com/datasets/rodolfofigueroa/spotify-12m-songs (Accessed: Dec. 07, 2023).

[7] Kang, E. B. (2023). On streaming-dating convergence: Music-mediated self-presentations on Tinder. New Media & Society, 25(8), 2057-2072. https://doi-org.ezproxy.library.dal.ca/10.1177/14614448211026618.

[8] J. S. Gulmatico, J. A. B. Susa, M. A. F. Malbog, A. Acoba, M. D. Nipas and J. N. Mindoro, "SpotiPred: A Machine Learning Approach Prediction of Spotify Music Popularity by Audio Features," 2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T), Raipur, India, 2022, pp. 1-5, doi: 10.1109/ICPC2T53885.2022.9776765.

[9] Harriman Samuel Saragih (2023) Predicting song popularity based on spotify's audio features: insights from the Indonesian streaming users, Journal of Management Analytics, DOI: 10.1080/23270012.2023.2239824