MELODY: Enhancing Music Recommendation with Large Language Models

Javier Austin Anak Jawa1, a), Hui-Ngo Goh1, b), Choo-Yee Ting1, c) and Albert Quek1, d)

1Faculty of Computing and Informatics, Multimedia University, 63000 Cyberjaya, Selangor Malaysia

*b) Corresponding author: hngoh@mmu.edu.my  
a)1211100857@student.mmu.edu.my*

*c)cyting@mmu.edu.my*

*d)quek.albert@mmu.edu.my*

**Abstract.** Recently, the rise of advanced machine learning techniques such Large Language Models (LLMs) has shown significant impact on music recommendation systems. Generally, Large Language Models (LLMs) are trained to understand complex user or item characteristics to generate better and relevant recommendations. This research explores and experiments on the common techniques used in music recommendation systems such as collaborative and content-based algorithms, with the integration of Large Language Model (LLM). A hybrid recommendation system will use K-Means for song clustering and BERT for generating song embeddings based on descriptions such as tags or metadata. Whisper-AI is utilized for transcribing song audio to lyrics for LLaMa and Phi-3 song analysis. The purpose of this song analysis is to assess how effectively the LLaMa and Phi-3 models interpret lyrics to improve the accuracy of song recommendations.

# Introduction

People has considered that music is an important aspect of their lives and they listen to music, an activity they engage in frequently [1]. Music streaming platforms like Spotify, Apple Music, and YouTube use data-driven approaches to provide curated song recommendations to their users. Recommendation typically uses collaborative and content-based filtering to predict user preferences [2],[3]. Collaborative filtering recommends items based on similar users' preferences, while content-based filtering uses item features the user already likes. A hybrid approach aims to combine both techniques for improved accuracy and relevance in music recommendation systems.

The scope of the work involves using the music dataset to explore patterns and generating useful recommendations by incorporating Large Language Models (LLMs) to understand natural language descriptions of music, user sentiments, and trends. The scope of the research project is limited exclusively to English song and lyrics. The reason being is to avoid potential challenges related to multilingual analysis and ensure consistency in textual data processing and language-based embedding.

The objectives of this research are to develop a hybrid music recommendation system by integrating K-Means clustering and BERT-based song embeddings for improved recommendation accuracy using song metadata and evaluating the effectiveness of LLaMa and Phi-3 models in analyzing transcribed song lyrics for enhancing recommendation relevance in music streaming platforms.

# literature review

Scalability and accuracy are major concerns in music recommendation, proposing collaborative filtering, content-based filtering, and hybrid models incorporating emotion and context-based retrieval [1]. Similarly, concerns like large catalog sizes, short track durations, emotional factors, and sequential consumption, along with issues like the cold start problem and playlist continuation are addressed. They also utilized hybrid approaches with emotion recognition and contextual adaptation [4]. Systems often struggle with efficiency and lack real-time updates when more variables are introduced, advocating for algorithms capable of handling complex, dynamic data [5]. K-Means clustering is introduced to group users by listening preferences and used PCA to manage high-dimensional data from platforms like Spotify [6].

Deep neural networks is explored for content-based filtering, extracting features from audio and metadata, and proposed multimodal systems using CNNs and autoencoders [7]. Metadata analysis with NLP for lyrics are combined, leveraging KNN and cosine similarity to find similar songs [8]. KNN is integrated with the Weighted Product Method for a hybrid multi-criteria recommendation model [9]. A Two-Tower Neural Network and sentence transformers for embedding textual metadata is implemented along with LLM-generated song summaries which enhances recommendation quality [10]. The application of DistilBERT-based system combining collaborative and content-based filtering for semantic song recommendation is introduced [11]. The utilization of LLMs such as ChatGPT, ChatGLM, and Llama in recommendation systems highlights their growing influence in the field [12].

# Methodology

## Data Collection

The music dataset is obtained from Kaggle. “Music Info.csv” from the Million Song Dataset[[1]](#footnote-1) will be used in this research. Table 1 represents the full list of each feature in “Music Info.csv” dataset with its description for context.

**TABLE 1.** “Music Info.csv” feature description table

|  |  |
| --- | --- |
| **Feature** | **Description** |
| track\_id | The unique identifier for each song. |
| name | The name of the song. |
| artist | The musical artist. |
| spotify\_preview\_url | The URL for Spotify preview of the song. |
| spotify\_id | The Spotify ID of the particular song. |
| tags | The list of tags describing the song. |
| genre | The genre of the song. |
| year | The year of release. |
| duration\_ms | The duration of the song in milliseconds. |
| danceability | The measure of how suitable the song is for dancing. |
| energy | The measure of the song intensity. |
| key | The musical key of the song. |
| loudness | The loudness of the song in decibals. |
| mode | The mode of the song. E.g., major or minor |
| speechiness | The measure of the presence of spoken words in the song. |
| acousticness | The confidence measure of the accousticness of the song. |
| instrumentalness | The measure of whether the song is instrumental. |
| liveness | The measure of the likelihood if a song is a live performance. |
| valence | The measure of the musical positivity. |
| tempo | The tempo of the song in BPM. |
| time\_signature | The estimated time signature. |

## Data Preprocessing

During data preprocessing, handling missing and duplicate values ensures dataset accuracy before EDA and model training. No duplicate rows were found in the music dataset. For missing values appeared in the “tags” and “genre” columns, since both columns contain similar genre information, missing values in one were filled using data from the other. Rows with missing values in both columns were removed, otherwise, the original data were retained.

## Song Clustering Model

The K-Means algorithm is applied for song clustering. Song is clustered based on numerical features such as “energy”, “danceability”, “tempo”, “loudness”, and this feature relevance is assessed using Explained Variance from Principal Component Analysis (PCA). The selected features are normalized before clustering. The optimal number of clusters (K) is determined using the elbow method. Once K is selected, K-Means is applied, and the resulting clusters are visualized via scatterplot. Clustering performance is evaluated using the Silhouette coefficient, Davies-Bouldin index, and Calinski-Harabasz score.

## Song Feature Embedding Model

The BERT model, bert-base-uncased[[2]](#footnote-2) is used to generate embeddings from the “tags” feature which contains descriptive metadata about songs. The BERT tokenizer processes the text into tokens, and the model generates contextual embeddings, unaffected by letter casing. Embeddings for approximately 50,000 rows are generated and saved in a NumPy (.npy) file for efficient storage and reuse, avoiding repeated computation. These song “tags” embeddings are then normalized, and cosine similarity is computed using the generated embeddings to recommend similar songs based on a user’s input song. Then, average cosine similarity is computed to evaluate each song in the list of recommended songs with the user's input song.

## Hybrid Approach (Song Clustering + Song Feature Embeddings) Recommendation

The implementation of a hybrid algorithm aims to further enhance the accuracy, relevance, and personalization of music recommendations. This hybrid approach leverages the strengths of both methods, (1) the BERT model’s ability to generate highly contextualized embeddings for textual metadata and (2) the K-Means algorithm’s effectiveness in clustering songs into well-defined groups based on musical feature similarity. The hybrid song recommendation algorithm generates recommendations by first identifying songs within the same cluster, and then applying a cosine similarity algorithm to compute the similarity index based on the generated song "tags" embeddings, ultimately recommending songs that are similar to the user’s input song. Evaluation is performed by computing the cosine similarity score of each song in the list of recommended songs with the user's input song.

## Audio-to-Lyrics and Song Analysis Model

# The Phi-3 and LLaMa language models are evaluated for their effectiveness in song analysis. Lyrics are transcribed using Whisper AI, a speech-to-text model, and then used as input prompts for Phi-3 and LLaMa. Users can also input custom prompts to explore song themes, moods, genres, or to generate descriptive tags for music recommendation. Model performance is assessed using a semantic similarity score, which measures how closely the generated analysis aligns with the original lyrics, offering a quantitative metric for the relevance and accuracy of the outputs.

# results and analysis

## Data Preprocessing

Table 2 shows the before and after data preprocessing of “tags” and “genre” column.

**TABLE 2.** "tags" and "genre" column before and after data preprocessing

|  |  |
| --- | --- |
| **Before Preprocessing of “tags” and “genre” column** | **After Preprocessing of “tags” and “genre” column** |
| A black and white screen with white text  Description automatically generated | A black and white screen with white text  Description automatically generated |
| Missing values found in “genre” column. Rows with a genre will retain their original genre. | Missing values of “genre” column filled with the first word of the “tags” column to represent the song’s genre. |

## Song Clustering Model

Figure 1 represents the graph of Explained Variance by Principal Component. The x-axis represents the principal component of numerical features in the dataset sorted from component 1 to 9 in order of loudness, energy, valence, speechiness, tempo, danceability, liveness, duration\_ms, and instrumentalness. The numerical features with the highest variance are loudness, energy, and valence, indicating greater variability in these audio characteristics across songs, while features like instrumentalness and duration\_ms show negative or low variance, suggesting limited or inverse variation in the dataset. Table 3 shows each of the numerical feature contributions to the first principal component in the data sorted in descending order.

A graph with a line and numbers

Description automatically generated

**FIGURE 1.** Explained variance by principal components

**TABLE 3.** Feature contribution to the first principal component

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Numerical Feature** | **Variance** |  | **Numerical Feature** | **Variance** |
| loudness | 0.557360 |  | danceability | 0.163492 |
| energy | 0.539118 |  | liveness |  |
| valence | 0.351041 |  | duration\_ms |  |
| speechiness | 0.220118 |  | instrumentalness |  |
| tempo | 0.215033 |  |  |  |

For better song clustering interpretation, features like “energy” and “danceability” reflect the overall feel or mood of a song, a new feature called “vibe” can be created to better capture the song’s liveliness and rhythmic character for clustering. Features like “tempo” and “loudness” together define a song’s “intensity”, a feature called “intensity” is created to represent how dynamic or energetic a track is for improved song clustering. Next, the elbow method graph is plotted to determine the appropriate K value, and the optimal value of K is 3. Therefore, the K-Means algorithm will be trained according to 3 optimal clusters. Figure 2 represents the scatterplot of K-Means Clustering of Songs with 3 optimal clusters.

A diagram of a number of dots

Description automatically generated

**FIGURE 2.** K-Means clustering of songs

Purple represents Cluster 0, turquoise represents Cluster 1, and yellow represents Cluster 2. Cluster 0 represents mid-energy, melancholic songs with some acoustic/instrumental elements, cluster 1 represents upbeat, danceable, and energetic songs, and cluster 2 represents low-energy, acoustic, or ambient-style music with a calm and mellow vibe. To evaluate the K-Means clusters, three types of evaluation metrics are computed. Silhouette coefficient, Davies-Bouldin, and Calinski and Harabasz score are used to evaluate the clustering. Table 4 shows the computed score value for the clustering based on the K-Means clustering of songs.

**TABLE 4.** Evaluation of metric scores for K-Means clustering of songs

|  |  |
| --- | --- |
| **Evaluation Method** | **Score** |
| Silhouette Coefficient | 0.38 |
| Davies-Bouldin | 0.89 |
| Calinski and Harabasz | 43404.05 |

According to Table 4, a silhouette score of 0.38 suggests that the data points are moderately clustered. However, this also indicates that the data points are slightly overlapping with another cluster, which means that the data points are close to the decision boundary of the neighboring clusters. A Davies-Bouldin Score of 0.89 indicates that the clusters are relatively well-separated and compact. A score below 1.0 is often considered good in practical clustering applications. However, a lower value closer to zero will be more ideal as that indicates the clusters are better-defined and distinct. A Calinski and Harabasz (CH) index of 43404.05 is relatively high. This indicates that the clusters are indeed well-separated and data points within each cluster are compact and closely related. The Calinski and Harabasz score is well reflected with the Davies-Bouldin score. Therefore, they suggest a strong clustering performance, with well-separated and compact clusters.

## Song Feature Embedding Model

Figure 3 shows the top 10 recommended songs based on the reference input song given by a user called Fix You[[3]](#footnote-3), by the popular British rock band, Coldplay. The recommended songs appear relevant by genre, and their accuracy is evaluated by computing the average cosine similarity scores based on the generated song embeddings between each recommended song and the user’s input song.

A screenshot of a computer screen

Description automatically generated

**FIGURE 3.** Recommended songs based on Fix You by Coldplay

The computed average cosine similarity score is 0.9867. A cosine similarity scores close to 1 means that the list of recommended songs and the song input by the user have high similarities. This suggests that the song embeddings generated by the BERT model can accurately recommend similar and relevant songs to users based on their input song characteristics.

# Hybrid Approach (Song Clustering + Embeddings) Recommendation

Figure 4 shows the comparison of the list of recommended songs by BERT embeddings with the list of recommended songs generated by the hybrid approach (song cluster + embeddings) song recommendation based on the input song, Fix You by Coldplay.

The hybrid song recommendation approach, which combines clustering and embeddings, suggests similar songs across more artists and genres compared to BERT-based recommendations. It captures diverse tracks like “The Tourist” by Radiohead from the reggae genre, with relevance supported by the average cosine similarity scores of 0.9860. Though the average cosine similarity score for the hybrid approach song recommendation is slightly lower compared to the average cosine similarity score for the BERT-based embeddings recommended songs, the list of top recommended songs generated by the hybrid song recommendation algorithm proved to have more variation of artists and it is also able to recommend songs across different genres. Hence, this encourages music listeners to explore, discover, and enjoy songs from diverse genres and artists that share similarities with their favorite artist.

|  |  |
| --- | --- |
| BERT-based Embeddings Recommended Songs: | Hybrid Approach Recommended Songs: |
| A screenshot of a computer screen  Description automatically generated  (a) | A screenshot of a computer screen  AI-generated content may be incorrect.  (b) |

**FIGURE 4.** Comparison of BERT-based embeddings recommended songs with Hybrid approach recommended songs

## Audio-to-Lyrics and Song Analysis Model

Song lyrics, transcribed from MP3 files using a state-of-the-art speech-to-text model, Whisper AI[[4]](#footnote-4), serve as prompts for Phi-3 and LLaMA models to analyze themes, mood, and sentiment for improved recommendations. The transcriptions are compared to ground-truth lyrics from Genius.com to assess Whisper AI’s accuracy. Table 5 shows the comparison of the transcribed lyrics by Whisper AI and the ground-truth lyrics referenced from “Genius.com” for the song, Beautiful Mess[[5]](#footnote-5) by the band, Diamond Rio.

**TABLE 5.** Transcribed lyrics by Whisper AI vs Ground-Truth lyrics from "Genius.com"

|  |  |
| --- | --- |
| **Transcribed Lyrics by Whisper AI** | **Ground Truth Lyrics from “Genius.com”** |
| Driving me wild What a beautiful mess, what a beautiful mess I'm in Spending all my time with you There's nothing else I'd rather do What a sweet addiction that I'm caught up in Cause I can't get enough, can't stop the hunger for your love What a beautiful mess, what a beautiful mess | Driving me wild What a beautiful mess What a beautiful mess I'm in Spending all my time with you There's nothing else I'd rather do What a sweet addiction that I'm caught up in 'Cause I can't get enough Can't stop the hunger for your love (What a beautiful) What a beautiful |

It is observable that Whisper AI model accurately transcribes the lyric from the song audio. This indicates that Whisper AI is reliable for generating high-quality text inputs for tasks like song analysis and recommendation.

The LLaMA model initialized for song analysis is llama-2-7b-chat.Q5\_0.gguf[[6]](#footnote-6) model and the Phi-3 model initialized for song analysis is the microsoft/phi-3-mini-4k-instruct[[7]](#footnote-7) model obtained from Hugging Face.

An example prompt for LLaMa and Phi-3 song analysis is shown in Table 6 and the resulting LLaMa and Phi-3 model song analysis response is shown in Table 7.

**TABLE 6.** Prompt for LLaMa and Phi-3 song analysis

|  |
| --- |
| **Prompt for LLaMa and Phi-3 Model Song Analysis** |
| You are a music expert. Read the following song lyrics and provide:  The main themes in 1–3 words each, the overall mood in one word, the sentiment (positive, negative, neutral), and suggested music tags for recommendation. Respond in paragraph format.  Lyrics: Driving me wild What a beautiful mess, what a beautiful mess I'm in Spending all my time with you There's nothing else I'd rather do What a sweet addiction that I'm caught up in Cause I can't get enough, can't stop the hunger for your love What a beautiful mess, what a beautiful mess |

**TABLE 7.** LLaMa and Phi-3 song analysis response comparison

|  |  |
| --- | --- |
| **LLaMa Song Analysis Response** | **Phi-3 Song Analysis Response** |
| The main themes in this song are love and obsession. The lyrics describe the speaker's intense feelings for the person they love, using words like "wild," "beautiful mess," "sweet addiction," and "hunger" to convey the depth of their emotion. The overall mood of the song is passionate and intense, with a driving beat and catchy melody that reflects the speaker's infectious energy. The sentiment of the song is overwhelmingly positive, with the speaker expressing their joy and gratitude for the love they've found. Music tags like pop, indie, and alternative suggest that the song could appeal to a wide range of listeners, while the adult contemporary tag implies that the song has a more polished and refined sound. | The main themes in these lyrics are love, obsession, and addiction. The overall mood is passionate. The sentiment expressed in these lyrics is positive, as the speaker is expressing love and addiction to someone. Suggested music tags for these lyrics could be "R&B", "Love Songs", "Soul", and "Obsession". |

Both responses by the LLaMa and Phi-3 model song analysis can capture the correct sentiment characteristics based on the lyrics of the song. Semantic similarity between the LLaMa and Phi-3 song analysis and Whisper AI transcribed lyrics is evaluated by embedding both texts using the all-MiniLM-L6-v2[[8]](#footnote-8) model sentence transformer, followed by measuring their similarity with the cosine similarity algorithm. Table 8 shows the semantic similarity score value of the transcribed lyrics by Whisper AI with the song analysis response by LLaMa and Phi-3.

**TABLE 8.** Semantic similarity score of transcribed lyrics by Whisper AI with the song analysis response by LLaMa and Phi-3

|  |  |  |
| --- | --- | --- |
| **Evaluation Method** | **LLaMa** | **Phi-3** |
| Semantic Similarity Score  (lyrics vs. models’ response) | 0.5943 | 0.4374 |

This indicates that the LLaMa model produced a song analysis response that is more semantically aligned with the original transcribed lyrics compared to the Phi-3 model.

# CONCLUSION

# This research highlights the potential of integrating machine learning and language models to enhance music recommendation systems. By combining K-Means clustering with BERT-based embeddings, a hybrid model is introduced that improves the diversity and relevance of song suggestions, helping users discover new artists and genres aligned with their preferences. Small and large language models like Phi-3 and LLaMa are also shown to support tasks like song analysis, each offering unique strengths based on task complexity and computational needs. Looking ahead, integrating generative AI into music platforms could revolutionize user experiences with features like personalized lyric analysis, mood-based suggestions, and AI-curated playlists. Future work should explore diverse datasets, including non-English lyrics, to improve inclusivity and recommendation quality.

# References

1. Song, Y., Dixon, S., & Pearce, M. A Survey of Music Recommendation Systems. *9th International Symposium on Computer Music Modelling and Retrieval (CMMR 2012)*, 395-405. (2012)
2. Z.B. Ter, N. Palanichamy, and J. J, “Generative AI-based Meal Recommender System,” *Journal of Informatics and Web Engineering* **4**(2), 315–338 (2025).
3. W.-E. Kong, T.-E. Tai, P. Naveen, and H.A. Santoso, “Performance Evaluation on E-Commerce Recommender System based on KNN, SVD, CoClustering and Ensemble Approaches,” *Journal of Informatics and Web Engineering* **3**(3), 63–76 (2024).
4. Schedl, M., Zamani, H., Chen, C. W., Deldjoo, Y., & Elahi, M. Current challenges and visions in music recommender. *International Journal of Multimedia Information Retrieval*, 95-111. (2018).
5. Fessahaye, F., Perez, L., Zhan, T., Zhang, R., Fossier, C., Markarian, R., . . . Oh, P. T-RECSYS: A Novel Music Recommendation System Using Deep Learning. *University of Nevada Las Vegas*. (2019).
6. Li, X. Analysis of machine learning-based music recommendation system using Spotify datasets. *International Conference on Software Engineering and Machine Learning*, 49-55. (2024).
7. Schedl, M. Deep Learning in Music Recommendation Systems. *Frontiers in Applied Mathematics and Statistics*, 1-9. (2019).
8. Tyagi, M. K., Ali, M., Kaim, G., Lamba, T., & Chugh, G. Music Recommendation System Using Multiple Machine Learning. *Internation Conference on Innovative Computing and Communication*. (2023).
9. Karishma, M. Music Recommendation System Using Machine Learning. *International Journal of Research Publication and Reviews*, 20-24. (2024).
10. Tekle, N., Ayala, A., Haile, J., Alshabanah, A., Baker, C., & Annavaram, M. Music Recommendation through LLM Song Summary. *The 1st Workshop on Risks, Opportunities, and Evaluation of Generative Models in Recommender Systems*. (2024).
11. Taief, A. M. Application of LLMs and Embeddings in Music Recommendation Systems. *Master thesis in Applied Computer Science*. (2024).
12. Liu, J., Liu, C., Zhou, P., Ye, Q., Chong, D., Zhou, K., . . . Yu, P. S. LLMRec: Benchmarking Large Language Models on Recommendation Task. (2023).

1. <https://www.kaggle.com/datasets/undefinenull/million-song-dataset-spotify-lastfm> [↑](#footnote-ref-1)
2. <https://huggingface.co/google-bert/bert-base-uncased> [↑](#footnote-ref-2)
3. <https://genius.com/Coldplay-fix-you-lyrics> [↑](#footnote-ref-3)
4. <https://openai.com/index/whisper/> [↑](#footnote-ref-4)
5. <https://genius.com/Diamond-rio-beautiful-mess-lyrics> [↑](#footnote-ref-5)
6. <https://huggingface.co/TheBloke/Llama-2-7B-GGUF> [↑](#footnote-ref-6)
7. <https://huggingface.co/microsoft/Phi-3-mini-4k-instruct> [↑](#footnote-ref-7)
8. <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2> [↑](#footnote-ref-8)