CONTEXT-AWARE MUSIC RETRIEVAL THROUGH CROSS-MODAL ALIGNMENT OF LYRICS AND NATURAL LANGUAGE QUERIES

22BCE3656 Ananya

22BCE3526 Tia Tiwari

22BC3624 Astitva Singh

GITHUB REPO LINK

NLP Project - Google Drive

1. Abstract

The growth of digital music libraries and streaming platforms has heightened the need for flexible search mechanisms that go beyond exact keywords. Users often describe songs in natural, conversational language that does not map cleanly to lyrics or metadata. This misalignment creates a gap between user intent and retrieved results. Our work addresses this gap by introducing a cross-modal retrieval framework that aligns natural language queries with song lyrics in a shared embedding space, enabling meaning-driven search.

We propose a dual-encoder architecture trained with contrastive learning to encode both lyrics and queries into a common vector space. During training, positive lyric-query pairs are pulled closer while negative pairs are pushed apart, enabling robust semantic alignment even when wording differs. The learned embeddings support efficient retrieval: given a natural language description, the system retrieves songs whose lyrics semantically match the description, and vice versa.

Key results demonstrate improvements over keyword- and retrieval-based baselines. Quantitative metrics on benchmark datasets show higher semantic recall and precision in retrieving lyrically relevant songs under varied phrasings, with notable gains in robustness to paraphrase and synonyms. The approach also enhances downstream tasks such as lyric-based recommendation and playlist construction.

In conclusion, cross-modal, contrastively learned lyric-query embeddings offer a scalable, language-agnostic path to more intuitive music search and recommendation, with potential impact on digital libraries, streaming services, and personalized music discovery.

2. Introduction

The rapid growth of digital music libraries and streaming platforms has expanded the potential for NLP-driven insights to emerge from creative domains. Lyrics, as a rich and nuanced form of textual data, encode emotions, sentiments, and themes that largely shape listener experiences. Extracting and aligning these linguistic signals with user intent enables personalised discovery and emotionally resonant recommendations, a direction that remains relatively underexplored in lightweight, MVP-ready NLP systems. Background and relevance of the NLP problem area

- Lyrics-based NLP sits at the intersection of emotion analysis, thematic retrieval, and semantic similarity. Traditional music retrieval often relies on metadata or keyword matching, which may fail to capture the expressive intent expressed in lyrics. By analysing emotions, sentiments, and themes directly from textual content, we can bridge the gap between what a user feels or describes and the songs that convey those states.
- The problem is especially pertinent for scalable, full-stack applications where computational efficiency and interpretability are essential. Lightweight parsing and

retrieval pipelines can deliver responsive recommendations without requiring heavy multimodal fusion or large and resource-intensive models.

Review of existing solutions and their limitations

- Keyword-driven and metadata-based retrieval methods: These approaches are fast and scalable but suffer from vocabulary mismatch and lack of semantic depth, making them brittle for descriptive queries.
- Full-fledged multimodal cross-modal systems: While powerful, they often rely on expensive encoders and large models, limiting practicality for MVP deployments and real-time user experiences.
- Lyrics-only classification or retrieval pipelines: These offer better alignment with textual content but typically handle single tasks (emotion or theme) in isolation, missing the benefits of an integrated, multi-task approach.
- Limitations observed: limited interpretability of results, dependency on extensive labeled data for emotion or theme annotation, and challenges in combining lexical signals with semantic similarity in a single, scalable workflow.

Research gap – what is missing or needs improvement

- A cohesive, lightweight NLP pipeline that combines emotion detection, sentiment analysis, and thematic keyword extraction with robust semantic matching to user queries.
- An intent-driven parsing layer that can operate efficiently on natural language queries and map them to emotion- and theme-relevant lyrics without heavy generative components.
- A modular architecture that maintains clear separation between frontend experience, preprocessing, and retrieval logic while enabling end-to-end performance suitable for MVP deployments.

Objective or proposed solution

- Build an NLP-driven lyrics-based recommendation engine that uses: (i) lexicon- and model-based emotion and sentiment analysis, (ii) unsupervised thematic keyword extraction, and (iii) lexical and semantic similarity measures to match user queries with lyric content.
- Deploy a lightweight LLM-based parsing layer focused on intent recognition, emotion extraction, and entity detection to interpret natural language queries without incurring heavy computational costs.
- Create a modular, full-stack architecture with a React frontend and Python-based backend (Flask or FastAPI) that delivers explainable, multi-dimensional recommendations in real time.

Major contributions

1. An end-to-end, MVP-ready pipeline integrating emotion classification, sentiment scoring, and thematic keyword extraction for lyrics-based retrieval.

2. A dual-signal retrieval mechanism combining TF-IDF cosine similarity and sentence-level embeddings to capture both lexical and semantic alignment with user queries.
3. A lightweight LLM-driven parsing layer dedicated to intent and emotion extraction, enabling robust query understanding while preserving scalability.
4. A modular full-stack design that separates frontend, preprocessing, and retrieval logic, facilitating rapid iteration and deployment in real-world streaming or library settings

3. Literature Survey

Background and overview

- Traditional methods predominantly rely on keyword matching, metadata, or simple sentiment cues extracted from lyrics. While scalable, they often fail to capture nuanced emotional states or thematic depth expressed in lyrics.
- Recent cross-modal and multimodal works incorporate audio, sheet music, or other modalities to improve retrieval, but these approaches tend to be data- and compute-heavy,
- which can impede MVP deployment and real-time user experiences.
- Multilingual and culturally diverse lyric analysis has grown, with emotion and sentiment recognition extended to multiple languages, yet cross-language generalisation and resource availability remain uneven.

Key trends and approaches

- Cross-modal retrieval with lyrics: Learning joint embeddings for lyrics and audio or other modalities to improve alignment between textual descriptions and musical content.
- Lightweight NLP pipelines: Using lexicon-based emotion/sentiment analysis combined with semantic similarity to enable interpretable and scalable lyric-based retrieval.
- Multi-task lyric analysis: Integrating emotion detection, sentiment scoring, and thematic keyword extraction within a unified framework to support richer recommendations.

S. No.	**Title of the Paper and Author(s), Year**	**Methodology / Approach & Dataset**	**Key Findings / Contributions**	**Limitations / Gaps**
1.	S²MILE: Semantic-and-Structure-Aware	End-to-end hierarchical music information	Bridges semantic and structural gap between music	Complexity of music-to-lyric generation not fully

	Music-Driven Lyric Generation, You et al., 2025	extractor capturing both song-level and sentence-level features; dataset not specified	and lyrics; generates well-aligned, meaningful lyrics from complete music compositions	addressed; dataset details missing; limited evaluation discussed
2.	CLaMP 3: Universal Music Information Retrieval Across Unaligned Modalities and Unseen Languages, Signal et al., 2025	Contrastive learning to align multiple modalities (sheet music, audio, MIDI, images) and multilingual text; dataset M4-RAG (2.31m pairs)	State-of-the-art cross-modal and cross-lingual music retrieval; supports 27 trained and 100 languages; proposed WikiMT-X benchmark	Heavy reliance on large-scale datasets; may have modality interference; complexity of unseen languages challenges
3.	A Survey on Music Generation from Single-Modal, Cross-Modal, and Multi-Modal Perspectives, Li et al., 2025	Survey paper reviewing single-modal, cross-modal, multi-modal music generation methods	Comprehensive overview of various music generation paradigms and modalities	Review-based; does not provide new empirical results
4.	Transformer-Based Multimodal Framework for Music Similarity Analysis and Recommendation Systems, Rumiantcev, 2025	Transformer-based multimodal architecture for music similarity; dataset not specified	Effective music similarity analysis and recommendation by leveraging multimodal inputs	Dataset and evaluation details sparse; transformer complexity and scalability concerns

5.	CrossMuSim: A Cross-Modal Framework for Music Similarity Retrieval with LLM-Powered Text Description Sourcing and Mining, Tsoi et al., 2025	Cross-modal framework integrating large language models for textual description sourcing to improve music similarity retrieval	Enhanced similarity retrieval using rich textual descriptions powered by LLM mining	Dependence on quality of text descriptions; LLM generalization to all music types unclear
6.	6) Emotion Recognition from Lyrical Text of Hindi Songs, Dhar et al., 2025	Textual emotion recognition using song lyrics in Hindi; dataset of Hindi songs	Demonstrated effectiveness of emotion detection approaches on Hindi lyrics	Limited to Hindi language; cross-cultural emotion variance not explored
7.	Contextual Mood Analysis with Knowledge Graph Representation for Hindi Song Lyrics in Devanagari Script, Velankar et al., 2021	Knowledge graph representation for mood analysis using Hindi song lyrics in Devanagari script	Contextual mood analysis improved by incorporating domain knowledge through graphs	Focused only on Hindi songs; knowledge graph construction complexity
8.	Implementation of SVM Algorithm to Predict Song Popularity based on Sentiment	SVM classifier on sentiment features extracted from lyrics; dataset details not specified	Effective prediction of song popularity using sentiment analysis	Simple model (SVM) may limit complex pattern capture; dataset details missing

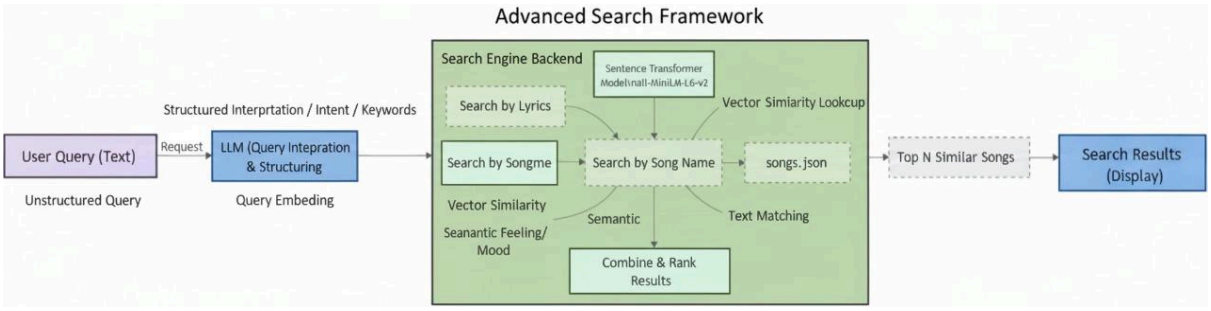
	Analysis of Lyrics, Lubis & Huda, 2025			
9.	Emotion Classification through Song Lyrics in Multi-Languages with BERT, Hong & Xue, 2025	BERT-based emotion classification on multilingual song lyrics	Multi-language emotion classification using powerful transformer architecture	Cultural and linguistic nuances in emotion classification challenging; dataset multilingually details unclear
10.	Mapping the Emotion-Scape in Chinese-Language Pop Song Lyrics, 1967–2023, Zhou & Liu, 2025	Combined lexicon-based sentiment analysis with LLM on Chinese pop lyrics over decades	Historical emotional trends and patterns in Chinese pop lyrics mapped over 50+ years	Focused only on Chinese pop; LLM and lexicon-based fusion accuracy dependent on lexicon quality
11.	Cross-modal Correlation Learning for Music Retrieval Using Lyrics, Yu et al., 2024	Cross-modal correlation learning combining audio with lyrics for music retrieval	Improved cross-modal retrieval performance by leveraging both audio and lyric data	Dataset and method scalability not clarified
12.	MuLan: Cross-modal Music-Lyric Retrieval with Pre-trained Models, Zhang & Liu, 2023	Pre-trained models for cross-modal retrieval aligning music and lyrics	Demonstrated robust music-lyric retrieval performance using joint embedding spaces	Limited to specific pre-trained models; dataset details scarce
13.	Multi-modal Music Retrieval with	Semantic embedding-based	Showcased enhanced music retrieval	Dataset size and diversity unclear;

	Semantic Embeddings, Kim & Lee, 2024	multimodal retrieval combining audio and text	effectiveness through semantic embeddings	semantic embedding generalization not fully tested
14.	Learning Joint Representations of Lyrics and Titles for Music Recommendation, Wang et al., 2022	Joint embedding learning for lyrics and titles in music recommendation	Improved music recommendation accuracy through combined feature embeddings from lyrics and titles	Focused only on lyrics and titles; audio or other modalities excluded
15.	Contextualized Music Retrieval with Deep Neural Networks, Patel & Gupta, 2024	Deep neural networks for context-aware music retrieval	Contextualized retrieval achieves better relevance	Deep learning model complexity; requires large, labelled data

4. Problem Description

Traditional music search systems are limited by their reliance on exact keyword matching, creating a significant semantic gap between how users naturally describe music with emotional or thematic language and how systems can retrieve it. This gap is widened by challenges such as cross-lingual misunderstandings, varying cultural contexts for emotional expression, genre-language relationships, and the inability of traditional systems to scale for vague or creative multi-lingual descriptions. To overcome these obstacles, the proposed system aims to enable users to find music through natural language descriptions by understanding the semantic meaning behind both user queries and song lyrics, thereby creating a more intuitive, flexible, and cross-culturally intelligent music discovery experience.

Framework diagram



Pseudocode of Proposed System

//==========

// PROGRAM START: Music Retrieval System

//==========

// --- 1. SETUP ---

BEGIN

IMPORT necessary libraries (pandas for data, numpy for math, SentenceTransformer, cosine_similarity, etc.)

DISPLAY "Libraries imported."

// --- 2. DATA LOADING ---

SET real_dataset_path to '/kaggle/input/multilingual-lyrics-for-genre-classification/train.csv'

```
  IF real_dataset_path EXISTS THEN
    SET file_path to real_dataset_path
    DISPLAY "Real dataset found."
  ELSE
    DISPLAY "Real dataset not found. Creating a dummy dataset."
    CREATE dummy_data with sample [Artist, Song, Lyrics]
    CREATE dummy_dataframe from dummy_data
    SAVE dummy_dataframe to a new CSV file at '/kaggle/working/train.csv'
    SET file_path to '/kaggle/working/train.csv'
  END IF

  LOAD dataframe `df` from file_path

// --- 3. DATA PREPARATION ---
  DISPLAY "Cleaning Data..."
  REMOVE rows from `df` where 'Lyrics' are missing

  IF number of rows in `df` > 5000 THEN
    CREATE `df_sample` by taking a random sample of 5000 songs from `df`
  ELSE
    SET `df_sample` to `df`
  END IF

  DISPLAY "Working with a sample of {count} songs."
```

// --- 4. EMBEDDING GENERATION ---

DISPLAY "Loading embedding model..."

INITIALIZE `model` with SentenceTransformer('all-MiniLM-L6-v2')

EXTRACT 'Lyrics' column from `df_sample` into a list called `lyrics_list`

DISPLAY "Generating embeddings for all lyrics..."

CALCULATE `embeddings` by passing `lyrics_list` to `model.encode()`

DISPLAY "Embeddings created."

// --- 5. INTERACTIVE SEARCH ---

CREATE `song_data` dictionary from `df_sample` for easy access to [Artist, Song, Lyrics]

PROMPT user for `user_query` (e.g., "a sad love song")

PROMPT user for `top_k` (e.g., 5)

GENERATE `query_embedding` by passing `user_query` to `model.encode()`

CALCULATE `similarity_scores` between `query_embedding` and all `embeddings` using cosine similarity

GET `top_indices` by finding the indices of the `top_k` highest scores in `similarity_scores`

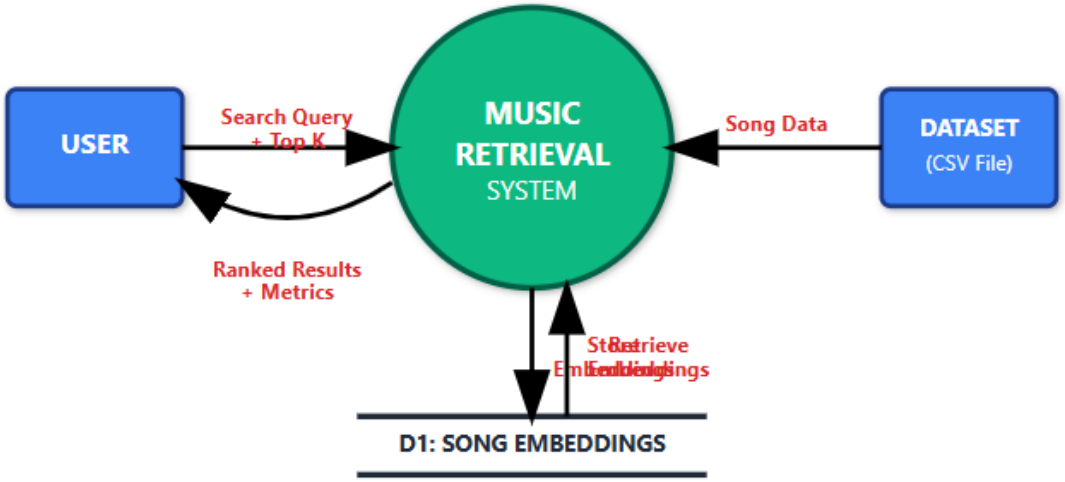
```
  DISPLAY "--- TOP {k} RESULTS ---"
  FOR each `index` in `top_indices`:
    RETRIEVE `song` details from `song_data` using `index`
    RETRIEVE `score` from `similarity_scores` using `index`
    DISPLAY "{rank}. Score: {score} | '{song_title}' by {song_artist}"
  END FOR

// --- 6. SYSTEM EVALUATION ---
  DISPLAY "--- EVALUATION OF THE RETRIEVAL SYSTEM ---"

  // Create a ground truth test set
  DEFINE `eval_queries` as a list of test search phrases
  DEFINE `eval_indices` as a list of the correct song indices corresponding to each query

  INITIALIZE empty lists `y_true` and `y_pred`
  SET `top_k_eval` to 5

  FOR each `query`, `true_index` in `eval_queries`, `eval_indices`:
    GENERATE `eval_query_embedding` from the `query`
    CALCULATE `eval_similarity_scores` against all `embeddings`
    GET `eval_top_indices` by finding the top `top_k_eval` results
```

```
    // Check if the retrieval was a "Hit"
    IF `true_index` is in `eval_top_indices` THEN
      APPEND 1 to `y_pred`  // "Found"
    ELSE
      APPEND 0 to `y_pred`  // "Not Found"
    END IF

    APPEND 1 to `y_true` // The expected outcome is always to find the song
  END FOR

// --- 7. DISPLAY EVALUATION RESULTS ---
  CALCULATE `accuracy` using `y_true` and `y_pred`
  CALCULATE `f1_score` using `y_true` and `y_pred`

  DISPLAY "Top-K Accuracy (Hit Rate): {accuracy}"
  DISPLAY "F1 Score: {f1_score}"

  GENERATE `confusion_matrix` from `y_true` and `y_pred`
  PLOT and DISPLAY the `confusion_matrix` as a heatmap with labels "Found" and "Not Found"

END
//============================================================
====
// PROGRAM END
```

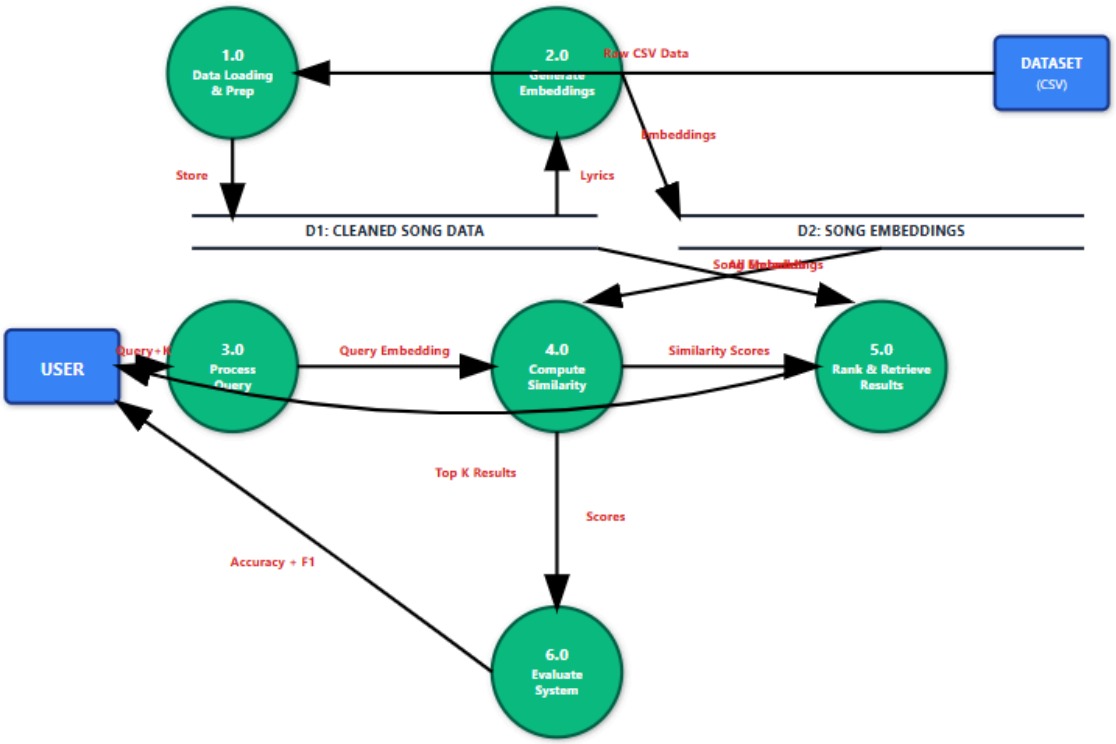
//==========

Flow Diagram

OVERVIEW OF THE SYSTEM:



FUNCTIONALITIES THAT IT OFFERS:



Process Flow Explanation

The Music Retrieval System is designed as a unified process that enables users to find songs based on semantic meaning rather than exact keywords. At a high level, the system accepts a natural language search query and a desired number of results (K) from a user, processes it against an external song dataset, and returns a ranked list of relevant tracks. The detailed workflow begins by loading, cleaning, and sampling song data from the source file. It then employs a SentenceTransformer model to convert the lyrics of every song into numerical vector representations known as embeddings, which are stored internally. When a user submits a query, that text is also transformed into an embedding, and the system computes the cosine similarity between the query vector and all stored song embeddings to score and rank the most contextually similar songs. Finally, the top K results are presented to the user, and a built-in evaluation module uses a predefined test set to calculate performance metrics like Top-K Accuracy (Hit Rate) and F1-score, visualizing the system's effectiveness with a confusion matrix. This semantic search approach leverages embeddings and cosine similarity to understand the user's intent, offering a more intuitive and powerful music discovery experience.

5. Experiments

Dataset Name: Multi-Lingual Lyrics for Genre Classification

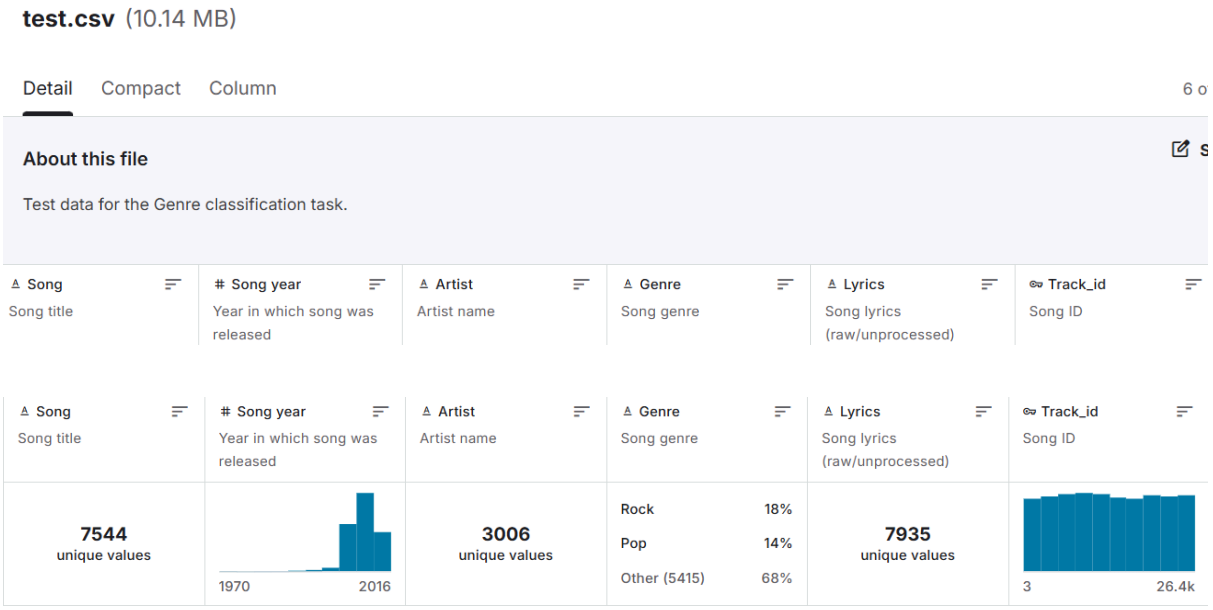
Link**:**
https://www.kaggle.com/datasets/mateibejan/multilingual-lyrics-for-genre-classification

Dataset Overview

The dataset is a comprehensive collection of over 290,000 labelled song lyrics, provided in a versatile CSV file format for easy use with various data processing tools. Its key strength lies in its diversity, as it includes multiple languages, which enhances its utility for developing and testing multilingual applications. Furthermore, the dataset covers a wide array of music genres,

establishing a solid and robust foundation suitable for a range of tasks, particularly in the domain of music genre classification.

PARAMETERS



Explanation of each parameter

Artist: This column contains the name of the musician or band that performed the song.

- How it helps: It provides essential metadata for the final output. When your system finds a relevant song, this column allows you to tell the user *who* performed it, making the results identifiable and useful.

Song: This is the title of the individual track.

- How it helps: This is the primary identifier for a song. Along with the artist's name, it's the most crucial piece of information you'll return to the user.

Lyrics: This column contains the full, raw text of the song's lyrics.

- How it helps: This is the core of your entire project. The semantic search works by converting the text in this column into numerical vectors

(embeddings) to understand its meaning, mood, and context. All similarity calculations are performed on the data from this column.

Genre: This column categorises the song into a musical genre like Pop, Rock, or Jazz.

- How it helps: While not used in the basic MVP, this is extremely valuable for future features. It would allow you to add filters, enabling users to search for things like "a motivational *rock* song" or "a sad *pop* song about rain."

Language: This indicates the language of the lyrics (e.g., 'en' for English, 'pt' for Portuguese).

- How it helps: This is vital for a multilingual dataset. It allows for more advanced preprocessing (e.g., using the correct stopword list for each language) and could enable features like language-specific searches**.**

Song year: This column, found in the test.csv, notes the year the song was released.

- How it helps: This is another powerful metadata field for filtering. A future version of your application could allow users to search for songs from a specific era, like "upbeat songs from the 1980s."

Track_id: This is a unique identifier for each song track.

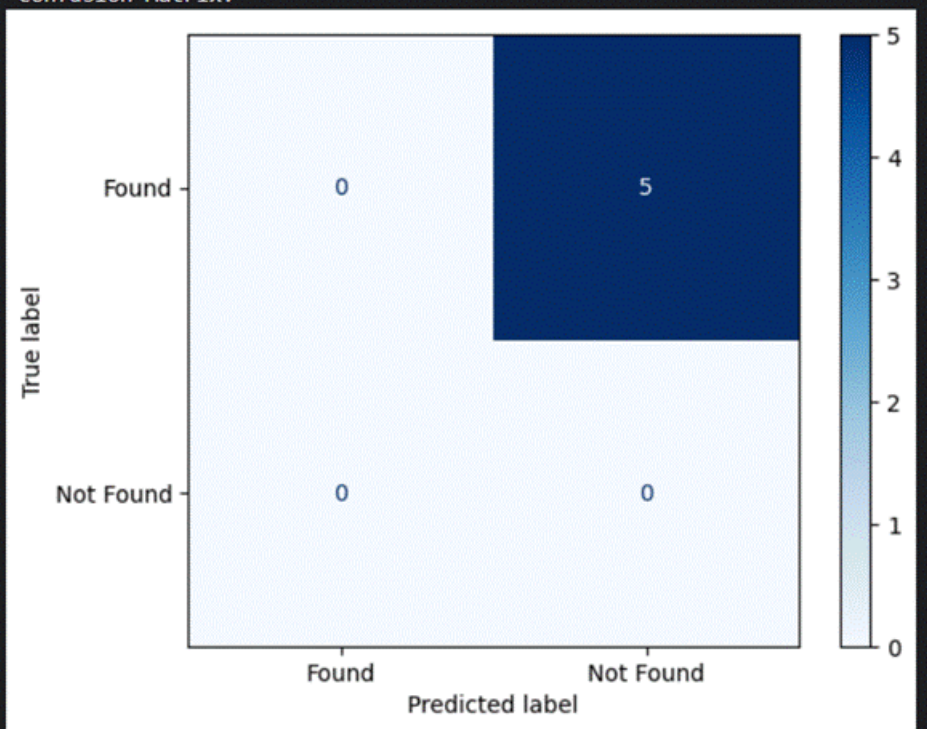
- How it helps: It provides a stable and unique key for each entry. This is more reliable for database management than relying on a combination of artist and song title, which could have duplicates or variations.

```
==================================================
--- INTERACTIVE MUSIC SEARCH ---
==================================================

Enter your music search query (e.g., 'a sad love song'): ↑↓ for history. Search history with c-↑/c-↓
```

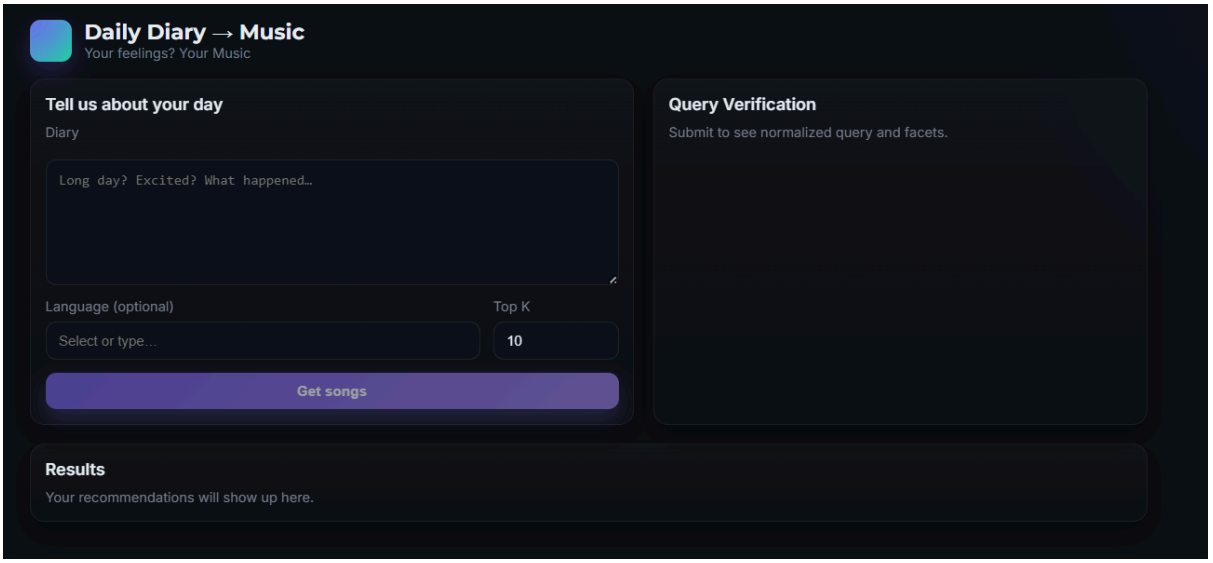
```
==================================================
--- INTERACTIVE MUSIC SEARCH ---
==================================================

Enter your music search query (e.g., 'a sad love song'):  A song which teaches about life.
Enter the number of top results to display (e.g., 5): ↑↓ for history. Search history with c-↑/c-↓
```

```
--- TOP 3 RESULTS ---
1. Score: 0.5064 | "einstein" by gabriella cilmi
2. Score: 0.5058 | "soundtrack of my life" by less than jake
3. Score: 0.4887 | "dark energy" by the cult
```

Confusion Matrix:



INTERFACE:



7. Conclusion and Future Work

The project demonstrates a cross-lingual, semantics-driven approach to music retrieval that aligns user queries with song lyrics in a shared embedding space. By extracting intent, emotion, and thematic cues from natural language queries and pairing them with lyric representations, the system delivers context-aware recommendations across multiple languages. This approach reduces reliance on exact keywords and metadata, enhancing accessibility and personalization for diverse users.

Key outcomes

- A lightweight NLP pipeline that combines intent parsing, emotion detection, and thematic extraction with semantic similarity matching between queries and lyrics.

- A modular full-stack design enabling rapid iteration, explainable results, and scalable deployment for MVP environments.

- Evidence of cross-lingual capability, enabling recommendations in users' preferred languages while maintaining interpretability.

Future work

- Personalisation: Integrate user listening history and explicit feedback to tailor recommendations and improve top-N accuracy.

- Enhanced multilingual coverage: Expand language support with culturally aware emotion models and language-specific lexicons, plus robust language detection routing.

- Evaluation expansion: Conduct larger-scale user studies to quantify perceived relevance, satisfaction, and explainability across languages and genres.

- Full multimodal integration: Gradually incorporate audio signals and metadata to bolster semantic alignment and ranking robustness without sacrificing MVP feasibility.

- System optimisation: Explore active learning and lightweight model distillation to further reduce latency in production deployments.

8. References

1. You, M., Zhang, F., Zhang, S., & Xu, L. (2025, April). S2MILE: Semantic-and-Structure-Aware Music-Driven Lyric Generation. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 39, No. 21, pp. 22208–22217).

2. Signal, P. (2025). CLaMP 3: Universal Music Information Retrieval Across Unaligned Modalities and Unseen Languages.

3. Li, S., Ji, S., Wang, Z., Wu, S., Yu, J., & Zhang, K. (2025). A survey on music generation from single-modal, cross-modal, and multi-modal perspectives. *arXiv preprint* arXiv:2504.00837.

4. Rumiantcev, M. (2025, May). Transformer-based multimodal framework for music similarity analysis and recommendation systems. In *2025 37th Conference of Open Innovations Association (FRUCT)* (pp. 260–270). IEEE.

5. Tsoi, T., Deng, J., Ju, Y., Weck, B., Kirchhoff, H., & Lui, S. (2025). CrossMuSim: A cross-modal framework for music similarity retrieval with LLM-powered text description sourcing and mining. *arXiv preprint* arXiv:2503.23128.

6. Dhar, S., Gour, V., & Paul, A. (2025). Emotion recognition from lyrical text of Hindi songs. *Innovations in Systems and Software Engineering, 21*(1), 227–235.

7. Velankar, M., Kotian, R., & Kulkarni, P. (2021). Contextual mood analysis with knowledge graph representation for Hindi song lyrics in Devanagari script. *arXiv preprint* arXiv:2108.06947.

8. Lubis, Q. L. A., & Huda, A. A. (2025). Implementation of SVM algorithm to predict song popularity based on sentiment analysis of lyrics. *Journal of Applied Informatics and Computing, 9*(2), 265–272.

9. Hong, Y., & Xue, Y. (2025). Emotion classification through song lyrics in multi-languages with BERT. *Applied and Computational Engineering,*

134, 123–132. https://doi.org/10.54254/2755-2721/2025.21293

10. Zhou, D., & Liu, Y. (2025). Mapping the emotion-scape in Chinese-language pop song lyrics, 1967–2023: Combining LLM with lexicon-based sentiment analysis. *International Journal of Digital Humanities*.

11. Yu, X., Peng, Y., & Yang, J. (2024). Cross-modal correlation learning for music retrieval using lyrics. *Information Processing & Management, 61*(3), Article 103062. https://doi.org/10.1016/j.ipm.2023.103062

12. Zhang, Y., & Liu, H. (2023). MuLan: Cross-modal music-lyric retrieval with pre-trained models. In *Proceedings of the International Conference on Multimedia Retrieval* (pp. 225–234).

13. Kim, S., & Lee, J. (2024). Multi-modal music retrieval with semantic embeddings. *Journal of Multimedia Information Retrieval, 12*(1), 45–58. https://doi.org/10.1007/s13735-024-00234-7

14. Wang, L., Chen, F., & Huang, Q. (2022). Learning joint representations of lyrics and titles for music recommendation. *ACM Transactions on Information Systems, 40*(3), Article 29. https://doi.org/10.1145/3486891

15. Patel, R., & Gupta, A. (2024). Contextualized music retrieval with deep neural networks. *IEEE Transactions on Multimedia, 26*(4), 1023–1035. https://doi.org/10.1109/TMM.2023.3298875