Logo, company name

Description automatically generated

**ICT619 Artificial Intelligence**

**Assignment 02**

**Music Recommendation system based on mood.**

**(MOOSIC)**

**Submitted By:**

**Manish Shakya (34509621)**

**Renuka Limbu (34591492)**

**Mohammad Ishrat E Hasan** (**34700886**)

# Abstract

Moosic is a ground-breaking method to customized music recommendation that takes use of the strong link between human emotions and musical preferences. This unique technology employs powerful machine learning algorithms to precisely anticipate a user's mood based on their interaction with music and other surrounding elements, resulting in a highly customized and emotionally resonant musical listening experience. It expands on existing recommendation systems by combining emotional intelligence, contextual awareness, and adaptive learning processes, guaranteeing that song recommendations are not just mood-appropriate but also contextually relevant. The system also includes Music player which allows the users to listen music on the platform itself or also the user can listen to the music on the Spotify Application, where the users will be directed.

# Introduction

Music, being a global language, powerfully connects with our emotions. In today's digital era, while people have access to an infinite number of music, choosing one that suits their mood remains difficult. This enormous selection might lead to decision paralysis. Furthermore, practically all systems recommend music based on the user's favourite songs or their overall listening habits, with little consideration given to the emotional component of song selection. To address this gap, our research introduces a Mood-Based Song Classifier and Recommendation System called Moosic. We employ machine learning to determine users' emotions based on their previous song choices, and then personalize suggestions to their emotional state. This not only gives a personalized listening experience, but also sheds light on the complex interplay between emotions and musical choices. We want to transform users' daily experiences with music platforms by ensuring that each song suggestion reflects their emotional journey. By analysing elements such as valence, acousticness, danceability, energy, instrumentalness, liveness, loudness, tempo, mood our system discerns subtle emotional cues within songs. We will also introduce an intuitive user interface that captures real-time feedback, fine-tuning our machine learning models to align recommendations with dynamic music selections and different mood states. (Torabi, 2023)

Moreover, the integration of contextual data such as time of day, weather, and personal schedules allows for the recommendation of music that not only matches the current mood but also complements the listener's environment, promoting an immersive auditory experience. Our research also contributes to the field by compiling a unique dataset, blending musical features with psychological profiles derived from user interactions, filling a niche in emotion-centric music recommendation. (Rokach, 2015)

By bridging the gap between vast music libraries and the individual's emotional needs, we anticipate a shift in how users interact with music streaming services. Our system aspires to be more than a platform—it seeks to be an empathetic companion that understands and enhances the listener's emotional well-being through music.

# Problem domain

* **Understanding Users' Preferences and Context**: Personalization and context awareness are critical. To increase relevance and user happiness, the system must precisely model individual preferences and tailor recommendations to the user's present position or activity.
* **Music Content Analysis and Semantic Gap**: Accurate music classification and recommendation need effective feature extraction from music files, as well as bridging the semantic gap between technical music features and user-perceived attributes such as mood or emotion.
* **Diversity in Recommendations**: Striking a balance between matching user tastes and providing fresh, different musical options is critical. This eliminates over-specialization of suggestions, allowing consumers to discover new music and artists while expanding their listening experience.
* **Evaluation and Feedback Mechanisms**: Measuring the success of mood-based recommendations is challenging. Traditional metrics like click-through rates may not fully capture user satisfaction in this context. The system needs mechanisms to gather feedback on the emotional impact of its recommendations and use this feedback to refine its models. (Schedl, 2018)

# Background and Motivation

The pursuit to harness machine learning ML for predicting users' emotions from their past music choices aims to enhance the music listening experience. By accurately gauging a listener's mood, the system can offer personalized song recommendations, enriching the user's emotional engagement with music. This initiative is driven by the clear link between a person's emotional state and their music preferences, aiming to develop a practical tool that optimizes daily musical interactions. Users can select their current mood with the selector which then will get them with their preferred recommendations. Also, as the users interact with the system by listening to the music, will auto update the recommendations automatically.

## Literature review and related works

* A music recommendation system using real time facial expression detections.

This paper describes the creation of a Mood-Based Music Recommendation System that employs machine learning to determine a user's mood based on real-time facial expression data. The system, which aims to increase consumer happiness by providing individualized song choices, is separated into two modules: mood recognition and recommendation.  
  
The mood detection module uses Keras' MobileNet model to categorize emotions from photos and works well with mobile devices due to its computational economy. The music recommendation module uses a song collection classified by mood and saved on Firebase to propose songs that fit the observed mood.

The Mood-Based Music Recommendation System was built utilizing machine learning techniques, notably the MobileNet model with Keras for real-time facial expression detection, and Java for Android app development. The solution deploys the trained ML model on mobile devices with TensorFlow Lite, uses Firebase for cloud storage and database services, and trains the model with datasets such as FER 2013 and MMA Facial Expression Recognition.

The suggested method detects mood with around 75% accuracy and offers music that may improve the listener's mood. Future enhancements might include adding physiological cues for increased accuracy and broadening suggestions beyond music to other types of entertainment such as movies or TV shows depending on identified mood. (Madik, A.)

* SVR-based music mood classification and context-based music recommendation by Seungmin Rho , Byeong-jun Han and Eenjun Hwang

The work "SVR-based Music Mood Classification and Context-based Music Recommendation" by Seungmin Rho, Byeong-jun Han, and Eenjun Hwang, published in October 2009, investigates a system meant to recommend music based on song mood categorization. This recommendation system combines many significant technology developments to propose music that is appropriate for the user's mood and surroundings.

The process involves classifying music moods using Support Vector Regression (SVR), which converts mood classification into a regression issue. The system classified music moods with an accuracy rating of 87.8% after utilizing SVR. The recommendation element makes use of both collaborative filtering and ontology technologies to determine the user's mood and situational environment before delivering suitable music recommendations. (Rho, S. (2009)).

**Existing System**

* **Lucyd**

It is a music recommendation engine created by four graduate students from UC Berkeley's Master of Information and Data Science (MIDS) program. Lucyd allows users to ask for music recommendations using their preferred phrases.

* **Reel time**

The AI system requires users to subscribe. Users may post photographs of huge groups, like retail malls, movie theaters, and restaurants. The algorithm distinguishes joyful and sad moods. The system distinguishes between joyful and sad expressions and assesses the scenario based on the faces of the participants.

* **Music.AI**

It suggests music based on user moods. It combines collaborative and content-based filtering approaches. Music selection considers emotion, time, atmosphere, and learning history.

# Methodology

## Dataset Description

We used Spotify Music data, previously we have used the dataset from <https://www.kaggle.com/datasets/musicblogger/spotify-music-data-to-identify-the-moods> which includes 686 songs from various artists and genres. For this final project we are using the larger dataset from Spotify [https://www.kaggle.com/datasets/vatsalmavani/spotify-dataset.](https://www.kaggle.com/datasets/vatsalmavani/spotify-dataset.%20) However, the dataset All song features are retrieved using the Spotify API. These metrics, such as pace, energy, valence, and danceability, shed light on the qualities and atmosphere of songs. The features indicated in the table below are Raw Features.

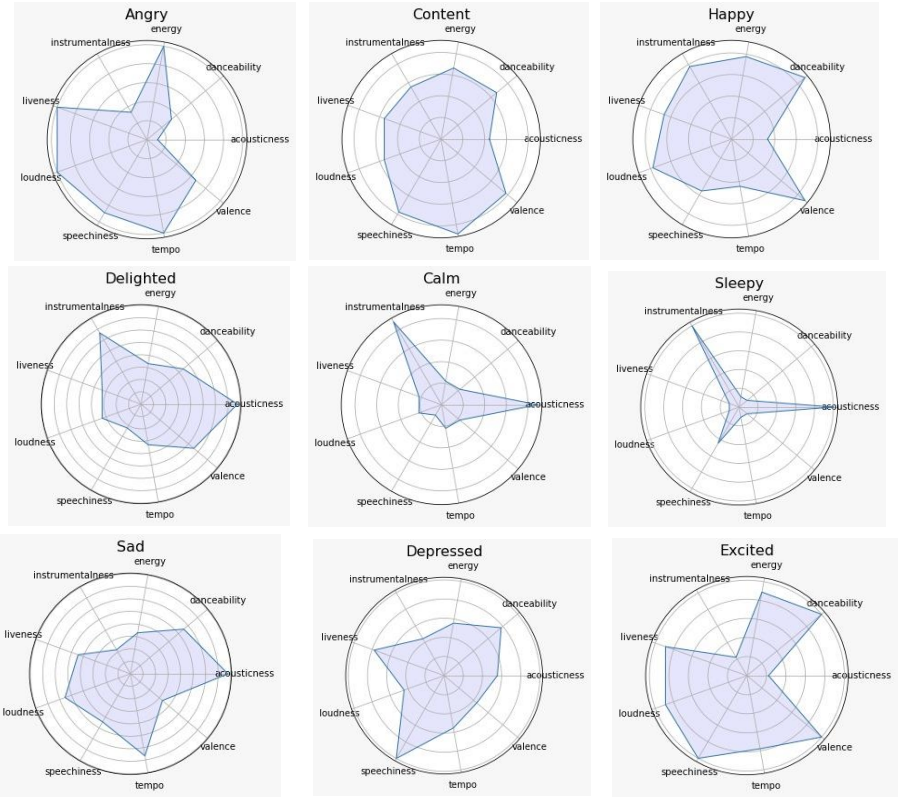
Some data pre-processing:

* Loading the datasets, the dataset we are using

Dataset 1: <https://www.kaggle.com/datasets/musicblogger/spotify-music-data-to-identify-the-moods>

Dataset 2: [https://www.kaggle.com/datasets/vatsalmavani/spotify-dataset.](https://www.kaggle.com/datasets/vatsalmavani/spotify-dataset.%20)

* How the moods are determined in the datasets



<https://www.irjmets.com/uploadedfiles/paper/issue_7_july_2022/27915/final/fin_irjmets1657529466.pdf>

Considering all the moods, we will be only choosing 4 different moods for this assignment.

A screenshot of a computer

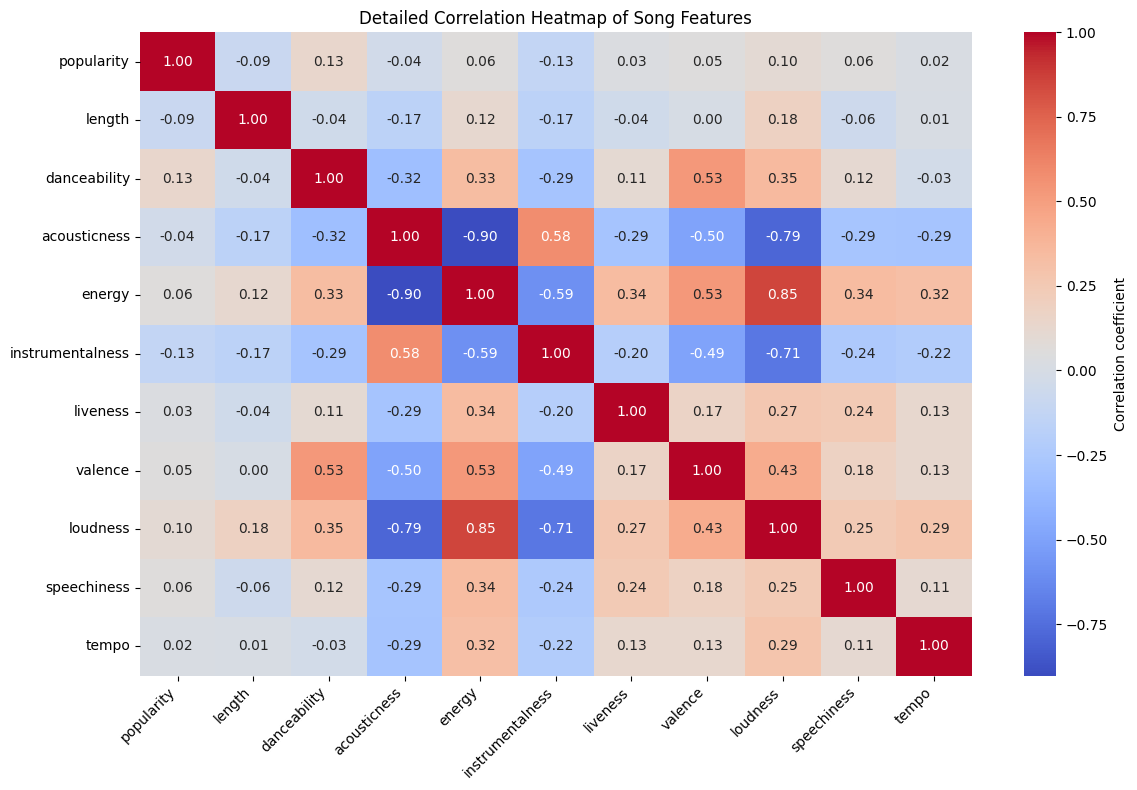
Description automatically generated

<https://towardsdatascience.com/predicting-the-music-mood-of-a-song-with-deep-learning-c3ac2b45229e>

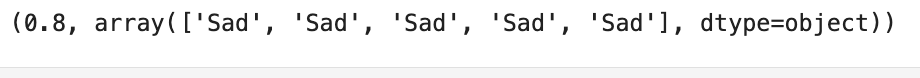
The table given above shows that there are some correlations between the features and the moods. Calm, music tends to have lower energy and danceability scores, and happy music tends to have higher speechiness, which might reflect the use of more positive words in the lyrics. Energetic musics have highest energy and dancebility scores also valence and loudness are greater.

* Adding mood column using the decision tree classifier from the trained based on previous dataset 1.

Based on the dataset we have created the features that are important using the correlation matrix



By analyzing the correlation heatmap we can see that the features that important are energy, acousticness, loudness, instrumentalness, valence. So based on that we will be using the decision tree classifier model created from the dataset 1 and add the mood column to dataset 2.



The accuracy of that comes out with 80%. We trained the data and applied that model to fill the mood column in the dataset 2. Exported the dataset as updated\_data.csv

* Check if any missing values, NA or duplicates.

A screenshot of a computer

Description automatically generated

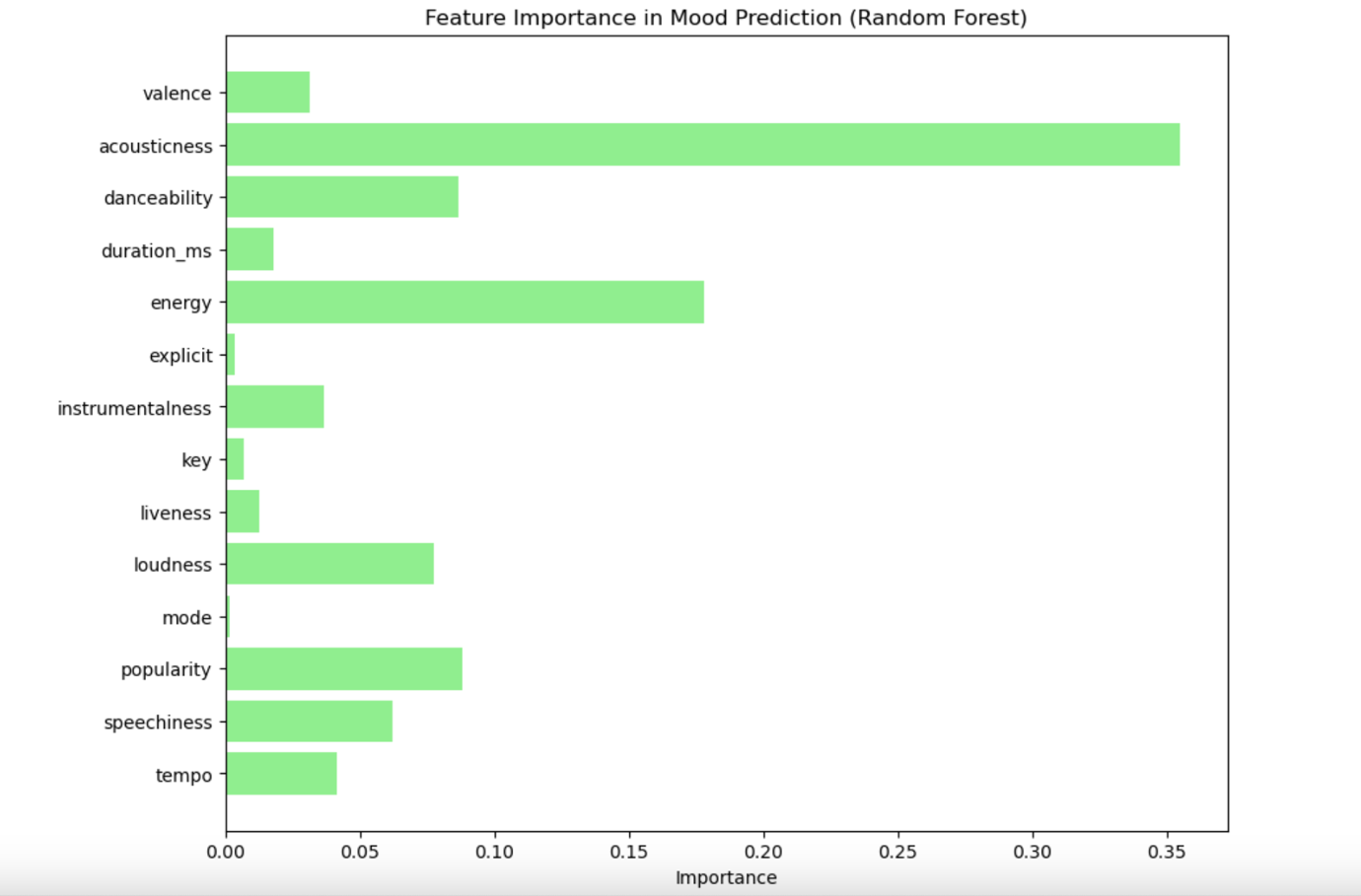
* The dataset 2 has 170653 rows and 20 feature columns.

|  |  |
| --- | --- |
| Feature | Description |
| name | The title of the track |
| album | The album where the track is from |
| artists | The artist who performed the track |
| id | A unique identifier for the track |
| release\_date | The release date of the track |
| popularity | A measure of the track's popularity |
| length | The duration of the track in milliseconds |
| danceability | A measure of how suitable a track is for dancing |
| acousticness | A measure of the acousticness of a track |
| energy | A measure of intensity and activity |
| instrumentalness | Indicates the likelihood that a track contains no vocals |
| liveness | Detects the presence of an audience in the recording |
| valence | Describes the musical positiveness conveyed by a track |
| loudness | The overall loudness of a track in decibels |
| speechiness | A measure of the presence of spoken words in a track |
| tempo | The overall estimated tempo of a track in BPM |
| key | The key the track is in |
| time\_signature | An estimated overall time signature of a track |
| mood | The mood classification of the track |

A screenshot of a computer

Description automatically generated

## Feature importance



Acousticness: This feature has the highest importance, suggesting that the acoustic properties of a song are highly predictive of its mood. This could mean that whether a song is more acoustic or electronic significantly affects its perceived mood.

Energy: Also, a major determinant of mood, indicating that the energy level (e.g., the intensity and activity of a song) strongly influences how its mood is perceived.

Danceability: Has significant importance as well, showing that the rhythm and tempo which make a song danceable are key indicators of its mood.

Loudness and Instrumentalness: These features, though less important than the top three, still considerably impact mood prediction. Loudness refers to the overall volume of a track, while instrumentalness measures the presence of vocal content.

## Exploratory Data Analysis

We will be generating some charts of important features to show, how we can use those features to generate a music recommendation. Below given some visualizations of those features

Distribution of Valence

A graph with a line

Description automatically generated

Valence is a measure of musical positivity, where higher valence indicates a more positive, happier track. This histogram shows that the distribution is somewhat uniform with slight increases toward the middle, suggesting a balance of tracks across the spectrum of emotional valence.

Distribution of Danceability

A diagram of a distribution of danceability

Description automatically generated

Danceability describes how suitable a track is for dancing based on tempo, rhythm stability, beat strength, and overall regularity. This graph shows a bell-shaped distribution centered around 0.6, indicating most tracks have moderate to high danceability.

Distribution of Energy

A graph of energy distribution

Description automatically generated

Energy is a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. The histogram shows a skewed distribution with most tracks having medium to high energy, peaking around 0.8.

Distribution of Loudness

A graph with a line

Description automatically generated

Loudness values are averaged across the entire track and measured in decibels (dB). This distribution shows a peak around -10 dB, indicating most tracks have a moderate loudness level.

Distribution of Tempo

A graph with purple lines

Description automatically generated

Tempo measures the speed of the music in beats per minute (BPM). The histogram reveals a bimodal distribution, with peaks around 60 BPM and 120 BPM, common tempos for many genres.

Distribution of Mood

A graph of different colored bars

Description automatically generated

This bar chart categorizes tracks into different moods: Sad, Happy, Energetic, and Calm. The chart shows that the dataset contains many Sad and Happy tracks, with fewer Energetic and very few Calm tracks.

## Classifications

With a new dataset we will be using different machine learning model to train our dataset. Below figure illustrates the accuracy of different models

A table of numbers with black text

Description automatically generated

There were several models like logistic regression, SGD classifier, Gaussian naïve bayes, decision tree, Random forest, XGB classifier, svm linear,With the above given figure we can say that the random forest stands out the most with 99.92% of accuracy.

# AI techniques

Based on mood selection by user:

We can see that the random forest stands out with 99.92% of accuracy. In our previous assignment we have used the XGB classifier, but comparing with the model accuracy of XGB classifier has 99.8% accuracy which is less than that of random forest. So, we will be using random forest to train our model.

Based on previous music selections:

We will be using a cosine similarity as cosine similarity is a useful tool for developing both types of recommender systems since it allows you to measure how similar persons, products, and content are. In this post, we'll utilize it to provide music suggestions based on how frequently people listen to certain songs. (Stieber, 2020) As user listen to music, using variables such valence, acousticness, danceability, energy, instrumentalness, liveness, loudness, tempo of that music the user listens to. The cosine similarity looks after the similar closest music which have similar or close data points of those variables.

# Evaluation Method

To evaluate our Mood based music recommendation system, we will creating a web application which involves, user interactions and recommend music based on those interactions. For that here are some given diagrams which shows how our system works.

* Use case diagram for music recommendation system.

A diagram of a person with a computer

Description automatically generated

* System flow diagram for music recommendation system.

A diagram of a software system

Description automatically generated

Start: The process begins when a user interacts with the system.

User Selects Desired Mood: The user chooses their current mood from available options, which informs the type of music recommendations they seek.

Generate Recommendations: The user clicks on a button to generate music recommendations based on the selected mood.

System Generates Recommendations: The system processes the input mood and generates a list of recommended songs that match the mood criteria.

User Listens to Music: The user listens to the music suggested by the system.

Decision to Listen on Spotify: If the user decides to listen to the music on Spotify, the system checks this choice.

Redirected to Spotify: If the user opts to listen on Spotify, they are redirected to the Spotify app where the recommended music is available.

End: The process ends, either after the user listens to the music or decides not to use Spotify.

* Tools to build this application.

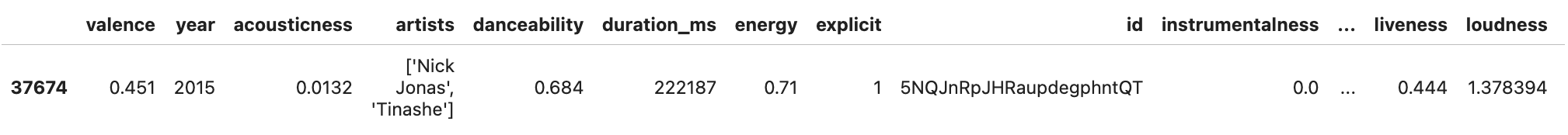
Flask, Random Forest classifier model, Cosine Similarity

* When we tried to filter out happy mood music, we change the mood encoder to 2 which will give us all the happy mood music.

A screenshot of a music list

Description automatically generated

* As user listens to music, the system uses cosine similarity to recommend the music.



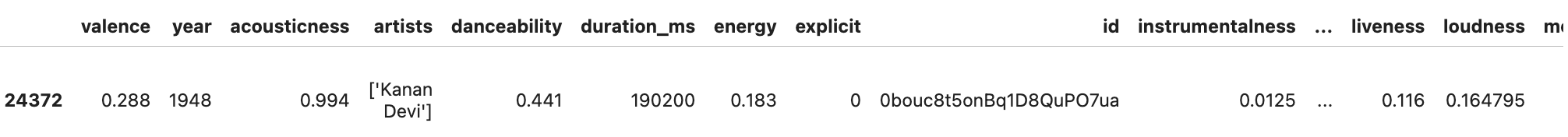
Lets assume that the user listen to this particular music given in figure 1.

The output of the cosine similarity, with 5 similar music recommendations.

A screenshot of a computer

Description automatically generated

We can see on the output that we get almost all 99.9 % of similarity in recommendations. However, again if we change the input, as



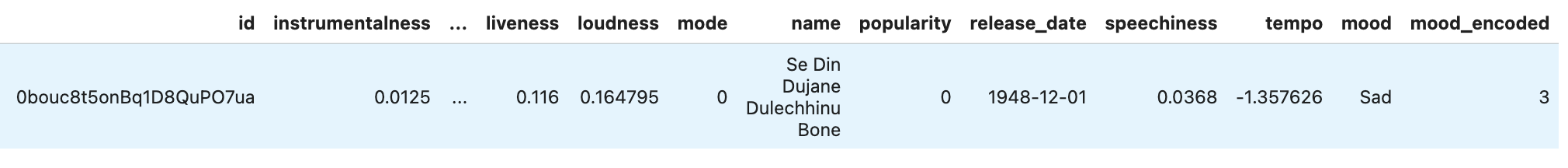
The output of the cosine similarity, with 5 similar music recommendations are as follows

A screenshot of a graph

Description automatically generated

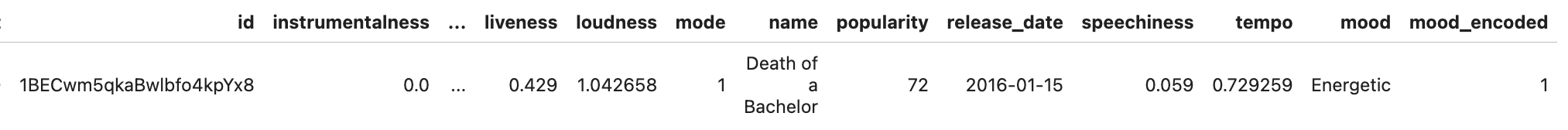
We can see that the similarity is in 99.95% really great, so we can assume that the cosine similarity is giving really well recommendations as the users listens to the music

# Results



A diagram of a song

Description automatically generated with medium confidence



A screen shot of a music chart

Description automatically generated

# References

Schedl, M., Zamani, H., Chen, C.-W., Deldjoo, Y., & Elahi, M. (2018). Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval*, *7*(2), 95–116. <https://doi.org/10.1007/s13735-018-0154-2>

Nguyen, L. V., Vo, Q.-T., & Nguyen, T.-H. (2023). Adaptive KNN-Based Extended Collaborative Filtering Recommendation Services. *Big Data and Cognitive Computing*, *7*(2), 106. <https://doi.org/10.3390/bdcc7020106>

Stieber, B. (2020, July 12). *Recommending Songs Using Cosine Similarity in R*. Brad Stieber. https://bgstieber.github.io/post/recommending-songs-using-cosine-similarity-in-r/#:~:text=Cosine%20similarity%20is%20helpful%20for