# A Self-Organizing Map Approach to Music Recommendation

**Catherine Mariana Philips**

Submitted September 2023, in partial fulfilment of the conditions of the award of the degree M.Sc. Data Science

School of Mathematics, Computer Science & Engineering
LIVERPOOL HOPE UNIVERSITY
Supervisor: Dr.Kapil Dev

I hereby declare that this dissertation is all my own work, except as indicated in the text

Date: 17/09/2023

# <u>DECLARATION</u>

This dissertation is submitted in partial fulfilment of the requirements for the degree of MSc offered by the School of Mathematics, Computing and Engineering, Liverpool Hope University.

I confirm that this dissertation is my own work and wherever required I have acknowledged the work of others.

I confirm that I have obtained informed consent from all people who participated in this work and received ethics approval as appropriate to my dissertation.

I confirm that the word count for this dissertation including the title page, declaration, abstract, acknowledgements, table of contents, list of illustrations, references and appendices is **‘11473’.**

I also permit Liverpool Hope University to store a copy of this dissertation in a public archive.

Catherine Mariana Philips

17/09/2023

## ABSTRACT

In this study, we suggest a novel approach for creating a music recommender that is especially suited for enthusiasts of K-pop. Our research focuses on using cosine similarity and Self-Organizing Maps (SOM) to build a reliable and effective recommender system. We methodically collect data using a multidimensional approach by utilising the Spotify API. In order to identify songs that are similar inside a two-dimensional grid, it is essential to group songs together based on their audio features. This is where the SOM algorithm comes into play. The results of the study show that the created model successfully recommends music based on user preferences.

An outcome of the developed model is the successful creation of a user-friendly interface called "MeloSOMmatic" which enables users to easily input their specifications and receive tailored recommendations. This innovative platform changes how users discover and appreciate Korean music by leveraging the power of SOM. Additionally, our work has effects that go beyond entertainment because it greatly advances the field of music technology by exhibiting the unique characteristics and diversity of the K-pop industry.

# ACKNOWLEDGEMENT

Without any support, completing this dissertation would not have been possible.

I want to start by thanking God for giving me the strength, guidance, and inspiration I needed to get through this dissertation.

Second, I would like to express my gratitude to my supervisor, Dr. Kapil Dev, for his consistent support, patience and understanding. This dissertation was completed successfully thanks to Dr. Dev's great help, prompt responses to countless emails, and readiness to answer all my questions.

Last but not least, I want to acknowledge the commitment and work I put into this project. My dedication and tenacity have been put on display throughout this journey, and I am proud of the contributions I have made.

## ABBREVIATIONS

SOM- Self-Organising Map

BMU- Best Matching Unit

RS- Recommendation System

# Table of Contents

## LIST OF FIGURES

# 1. INTRODUCTION

The digital era has brought about a fresh method of appreciating harmonies in the beautiful world of music, where melodies transcend boundaries and emotions find expression. Music lovers today have a dizzying array of musical options at their fingertips thanks to the growth of internet music platforms and streaming services [46]. Finding tunes that suit one's tastes while travelling through this sea of melodies might be challenging. This is where the wonderful symphony of music recommendation systems enters the picture, attempting to create tailored playlists that appeal to specific listeners [47].

## 1.1. Motivation

---

*“Language doesn’t matter to us that much, like in the past, we want to transcend everything, even ourselves. So, I guess that’s the power of music. In general, in language, topic, there’s no borders, boundaries, or limits.”*
*- Kim Namjoon*

---

Researching music recommendation algorithms captured my attention as a devoted music aficionado and an avid follower of the mesmerising K-pop phenomenon. The vivid visuals and melodious rhythms that define K-pop offer a blank canvas upon which we can envision how technology might elevate our musical encounters. The quest to comprehend how computers can interpret our unique musical nuances and curate customised playlists that transcend linguistic and cultural boundaries has ignited the inception of this project.

Although it sounds cliched, the global hit "Gangnam Style" was the beginning of my journey into the world of K-pop and the beginning of my fascination with this genre of music. K-pop's colourful visuals and contagious rhythms gave me a fascinating audio landscape and piqued my interest in the enormous impact of technology in the field of music.

The melodies of K-pop gave me comfort and motivation while I dealt with the pandemic's unparalleled hardships. The lyrics and harmonies evolved from simply being music to serving as a means of connection and healing following a life-altering incident. During this period, I reconnected with the K-pop scene and saw how it could inspire positivity and promote a sense of community even when people were separated by distance.

In a world where melodies evoke feelings devoid of language barriers and harmonies encourage connections that cross borders, this dissertation aims to combine the unifying power of music with cutting-edge technology. By exploring the complexities of music recommendation algorithms, we hope to create a symphony that integrates not only with the beats of our emotions but also with the various rhythms of the K-pop universe.

## 1.2. Background

*"Music is the universal language of mankind."*
*- Henry Wadsworth Longfellow*

Music has a unique capacity to transcend linguistic and cultural barriers and is frequently referred to as the universal language of emotions. The reason is that it exists in every society, including ones that use words and others that don't. It's interesting that it differs more among groups than within them and that it encourages particular behaviours. When used properly, music may help individuals connect with one another and with themselves [53].
In addition to inspiring improvisation, self-expression, and discovery, music fosters conditions where healing happens naturally.

There's no denying that music plays a crucial part in bringing consolation during difficult times with its capacity to give insight into our inner worlds and aid in the growth of resistance against life's hardships. Regardless of the genre, music has the power to mirror the human experience, provoke strong feelings, and unite people [54].
The global appeal of Korean pop culture, such as K-drama and K-pop music, is referred to as the "Hallyu wave." It has acquired popularity in several nations, particularly among the younger generation [55]. K-pop, or Korean Pop, is the most well-known, aesthetically centred, unconventionally designed, and trend-setting music genre of the twenty-first century. K-pop, which has its roots in South Korea, is influenced by a variety of musical genres, including pop, experimental, rock, hip-hop, R&B, electronic, and dance [56].

Early K-pop music was heavily influenced by the United States and Japan, and a plethora of musicians were playing "trot" music, which consisted of tunes adapted from British and American folk songs with Korean-language lyrics that, for the most part, objected to the country's colonial rulers. This was generally the norm at the time, mixed with ballads and slower country music [57]. The first generation of K-pop singers emerged once the recipe for K-pop popularity was uncovered (genre-defying music, beautiful performers, immaculate appearances, and precise choreography) [56].

However, most people will agree that BTS, one of the biggest South Korean sensations of all time, represents the pinnacle of K-pop's development. The chart-topping septet has made significant advancements for the K-pop industry as a whole by shattering records and stereotypes both domestically and abroad. Choosing to sing about more serious topics like relationships, societal pressures, and other narrative elements influenced by everything from Jungian philosophy to having a dream and following it, BTS has made it their business to adopt a model akin to the band that started it all, Seo Taiji and Boys [57].

Over time, there have been significant changes in how we listen to music. The development of music players has come a long way, from vinyl records to cassette tapes, CDs to MP3s, and now streaming services. From buying physical records or CDs to streaming music online, our method of consuming music has also changed [58]. Because they provide fair accessibility, music streaming services are growing in popularity among the younger population. For a monthly charge, you can make your own playlists and access millions of songs. Compared to buying individual albums or singles, this is significantly less expensive [59].

One can only expect more advancements in music streaming as technology keeps developing. With access to massive song libraries, services like Spotify, Apple Music, and YouTube Music have given users unprecedented convenience. Modern algorithms are used by these sites to create personalised playlists, suggest new music, and aid in music discovery, all of which improve the listening experience as a whole. The future of music consumption is certain to be even more engaging and interesting, with everything from improved audio quality to immersive virtual reality experiences [58].

### 1.3. Aims

This music recommendation system's main goal is to employ Self-Organizing Maps (SOMs) to their full potential to provide a user-friendly platform for song recommendations. The system's main goal is to make music suggestions for users based on their input, which includes genre preferences and different audio characteristics like danceability, energy, and more.

By making personalised song recommendations based on user preferences, this system seeks to improve the music-listening experience, with a special emphasis on the vibrant and varied world of K-pop music. The dissertation seeks to bridge the gap between users and their desired music by employing SOMs to decode complex musical patterns and correlations.

### 1.4. Objectives

This dissertation has multiple objectives, including the following:

- To create a useful system for recommending music that uses Self-Organising Maps (SOMs) to analyse input characteristics and spot patterns in music tracks.
- Apply preprocessing techniques to the input data, such as genre details and audio attributes, to alter and prepare them for SOM model training.
- To construct a low-dimensional representation of songs, train the SOM model using a large dataset of K-pop songs and their music genres.
- Calculate the similarity between the user's tastes and the training model to allow the SOM-based system to accurately recommend songs to the user.
- Check how well the recommendation algorithm is working to ensure the quality of the music suggestions.
- Examine how the SOM model responds to various musical genres and how the system may expand to handle a growing library of songs.
- Create a user interface that promotes user interaction by displaying the top music suggestions in an engaging and visually appealing manner, allowing users to easily explore the suggested songs.

With the help of these objectives, this dissertation seeks to create a platform for music recommendation that breaks the language boundaries, enhances users' musical experiences, and provides a seamless union of technology and music appreciation.

## 2. LITERATURE REVIEW

### 2.1. Recommendation Systems

Through the analysis of user, item, and user-item interaction data, recommendation systems (RS) leverage machine-learning techniques to provide item suggestions. The work presented by Fanca et.al primarily focuses on the development of various recommendation systems, their evaluation, and the utilisation of machine learning models to generate accurate ranking predictions [12].

Different types of recommendation systems employ both simple and complex algorithms to offer recommendations. Bhareti et al.'s study underscores the value of RS in enhancing interaction within applications, contributing to our comprehension of their role in enhancing user experiences [11].

Gironacci [14] underscores the significance of recommendation systems in revenue generation and the competitive advantage they offer to companies. Furthermore, the paper highlights how machine learning techniques enable the customization of recommendation systems. Notable examples of renowned companies implementing recommendation systems underscore their value in streamlining corporate processes.

Fouss' categorisation [13] divides RSs into four categories based on how preferences are collected, the approach taken, the algorithm employed, and the presentation of findings. This categorization aids researchers and practitioners in selecting the most suitable method based on their specific requirements.

Enhancements to existing recommendation systems are possible in several ways. For instance, exploring and developing more sophisticated algorithms for RSs using deep learning or reinforcement learning could enhance the accuracy and personalization of recommendations. Additionally, hybrid recommendation systems, which blend diverse algorithms or techniques to provide more diverse and accurate recommendations, offer another avenue [15].

### 2.2. Music Recommendation Systems

In recent years, the popularity of music recommendation systems has surged, leading to the development of a diverse range of methods and strategies. The demand for personalised recommendations has led to the emergence of user-centric systems that consider factors such as the user's background, emotions, and personality [8].

The significance of personalisation and user-centric system design has been emphasised by Kle and Wieczorkowska [7]. Deep learning techniques have gained prominence for their ability to leverage extensive datasets, enabling machines to make more precise predictions and thereby enhancing the accuracy and effectiveness of music recommendation systems [8].

Kumari and Khodhanpur [9] introduced a music recommendation engine that employs an item-based Collaborative filtering algorithm alongside the cosine similarity method. This engine assesses distinct musical attributes, such as genre, artist, album, language, length, and year, extracted from audio files. By analysing user preferences, their approach generates song suggestions. Their study primarily aimed to design a recommender system harnessing the advantages of item-based Collaborative filtering to deliver high-quality recommendations.

Much of the research in this domain has centred on devising methodologies to evaluate users' interactions with recommendations, often leveraging the diversity of reference items. However, only a limited number of studies have explored the connection between suggestions and the extensive spectrum of user attributes. These attributes span individual-level traits, like personal features, as well as group-level factors, including country of origin [8].

## 2.3. Music RS: Evolution and Significance

Several significant events in history have influenced the evolution of music recommendation. Automatic music suggestion is a current and active subject of research due to the introduction of music streaming services and the rise in music consumption [16]. Users now have trouble finding their favourite music due to an abundance of online music resources, prompting the employment of precise and effective music recommendations [17].

With the growth of music streaming services, the music industry is currently undergoing a transition that has made recommendation systems a crucial part of the business and created new issues like sequential and contextual recommendations [18].

Innovations in technology have had a significant impact on the evolution of music recommendation in several ways. First, due to the widespread use of the internet made possible by technological advancements, a vast amount of music content is now easily accessible to individuals everywhere [19]. Due to the readily available nature of music, research into automatic music recommendation is important and ongoing [16].

Secondly, recommendation systems have become crucial to music platforms, which are currently the leading source of income for the music industry due to the introduction of music streaming services and the rise in music consumption rates [20]. Lastly, the creation of music recommendation systems that can assess the mood or ambience of video or photographic data and suggest music that fits the user's surroundings is now possible thanks to technical advancements [21].

Systems for making personalised music recommendations based on user tastes and emotions have evolved throughout time to solve the problem of information overload. At first, content-based strategies were well-liked, using classification algorithms or deep learning methods to suggest music based on its characteristics or metadata [36].

There are various types of music recommendation systems. One type is the content-based filtering method, which recommends music based on the user's data and subjective qualities of the music such as speechiness, loudness, and acoustiness [24].

The way content-based music recommendations operate is by comparing the qualities of music tracks to the user's preferences. User input or collaborative filtering is not used in this method. Instead, it concentrates on the musical elements itself, such as acoustic characteristics and audio embeddings [37],[38]. The ability to compare music based on its audio content is made possible by machine learning algorithms like Siamese Neural Networks (SNNs) and deep learning methods [39].

Another form is the collaborative filtering method, which recommends content, based on user ratings and interactions [27]. In music recommendation systems, collaborative filtering is employed to make musical suggestions based on user preferences and shared characteristics with other users. It entails employing machine learning techniques like factorization machines to break down user interests and examine numerous elements that affect user behaviour [40].

The acoustic similarity-based content-based method, which analyses the acoustic aspects of musical compositions to make recommendations, is another approach [16]

## 2.4. Challenges and Advancements in Music RS

Systems for music recommendation have difficulties when using self-organising maps (SOM). One difficulty is the requirement to take the listener's context, such as mood and occasion, into account in addition to the genre or similarity of the audio [29]. Content-driven music recommendation faces six primary challenges: augmenting recommendation diversity and freshness, offering transparency and explanations, attaining context awareness, suggesting musical patterns, improving adaptability and productivity, and mitigating the "cold start" problem [10].

The future of research in the realm of music recommendation systems holds numerous challenges and opportunities. These encompass the enhancement of recommendation accuracy and personalisation, the resolution of privacy and security considerations, and the exploration of captivating development methodologies [8].

To learn the representation and distance measure between users and items for music recommendation, Liang et al. introduce the use of a triplet neural network that makes use of positive and negative samples. This interesting approach is initially trained using a loss function to minimise the distance between positive user-item pairs and maximise the distance between negative pairs. There are three branches in the network architecture: one for the user, one for the positive item, and one for the negative item. There are several layers of neural networks in each branch. Combining stochastic gradient descent and backpropagation methods, the triplet network is trained to maximise the network's parameters [41].

Rao et al. discuss investigating the use of collaborative filtering and content-based recommendation techniques to address the cold start problem and achieve high recommendation accuracy while also investigating the use of supervised feature learning for music recommendation in the context of music streaming services and the rise in music consumption [16].

Recommendation systems that offer a listening experience rather than just one-time recommendations are now required due to the present transformation in the music industry. These discoveries affect different facets of the ecosystem of the music industry and have resulted in advancements in context-aware music recommendation, automatic playlist production, and recommendations for the creative process of music-making [18]

## 2.5. Self-Organising Maps

Neural network models that are classified as unsupervised include Self-Organizing Maps (SOMs). They are used for large data processing processes, including reducing dimensions, clustering, and similarity finding [30]. For topological qualities to be discovered and projected, high-dimensional data must first be arranged into a 2-dimensional grid via SOMs. Their training phase can be computationally costly; however, a quick method has been suggested [31],[32].

Elend and Kramer's paper used filter banks of convolutional neural networks (CNNs) to enhance the quality of SOMs for high-dimensional data sets by using higher-level features from pre-trained convolutional layers. Their method calls on properly trained CNNs, which can be obtained from the same or similar domains, or in semi-supervised settings. [32]. The standard SOM algorithm can be improved by adding topology-related extensions, possibly increasing the analysis's accuracy and effectiveness [30].

The Self-Organizing Map (SOM) algorithm and its variations are frequently employed to tackle a wide variety of issues, despite the lack of strong mathematical justifications. Extensions to the kernel, dissimilarity, and categorical data have made them even more effective tools. The study also suggests that, as some SOM variations have more robust theoretical foundations, SOM may emerge as an easy representation of these robust algorithms [33].

SOMs are helpful for outlier detection and sampling because they can be understood as performing gradient descent and minimising an approximation to the log-likelihood of the Gaussian Mixture Model (GMM) [35]. Researchers and data analysts working with complicated datasets might benefit from the methods Kopczynska and Kopczynski provide in their study because they enable them to find hidden patterns, reduce dimensionality, and locate clusters or commonalities in the data [30].

Biological neurons served as the inspiration for the Dendritic Self-Organizing Map (DendSOM) algorithm presented by Pinitas et al., which outperforms traditional SOMs and cutting-edge continual learning algorithms. This suggests that neuronal features could be included in SOMs to overcome catastrophic forgetting [34].

### 2.6. Self-Organising Maps for Music RS

The technique of using SOMs for music recommendation includes building a network structure, feeding input data to train it, and then simulating the network to produce recommendations based on the trained model [22].

SOMs can be used in tandem with neural network-based techniques and cluster-based quotas to enhance the precision and effectiveness of music recommendation systems. Using the underlying popularity clusters to generate predictions that match the popularity distribution shown in the data, Wundervald's cluster-based quota system predicts music suggestions [23].

While maintaining the unique properties of each algorithm, this technique raises the recommendation frequencies of less famous artists. Sharma et al. recommend a different strategy based on neural networks where a person's facial expressions determine their mood, resulting in more effective and tailored music choices [24].

SOMs can use collaborative filtering approaches like user-based and item-based suggestions to assess the similarity of users and items [27]. SOMs can also be used for dimensionality reduction, which can be achieved by putting high-dimensional data in a lower-dimensional space while maintaining topological links using self-organising feature maps (SOM). Neighbourhood learning is used by SOM algorithms to produce a topological ordering of the data points, preserving closeness and distance relationships in the lower-dimensional space [48].

Systems for recommending music that use Self-Organizing Maps (SOMs) have both benefits and drawbacks. One benefit is that SOMs can effectively recommend songs by capturing the complex relationships between music tracks and representing them in a low-dimensional space [25]. SOMs can also manage enormous volumes of data and adapt to changes in consumer tastes over time [26]. For music recommendation systems, self-organizing maps (SOM) have been found to beat existing recommendation algorithms. Zhou et al.'s incremental-input SOM is economical and time-saving since it enables online updating of the recommendation model without retraining [50].

SOM's music recommendation algorithms have flaws. One negative is that it might not comprehend the preferences of the new user, which could cause a cold-start issue. Systems frequently use a random selection of songs to collect user preferences to remedy this, although this can take time [51],[52].

## 2.7. Summary

In the first section, titled ‘Recommendation Systems’, a brief description of music recommendation systems is given. Machine learning techniques are used by recommendation systems (RS) to provide item suggestions based on user interaction data. They improve user experiences, bring in money, and provide businesses with a competitive edge. On the basis of preferences, strategy, algorithm, and presentation, RS can be divided into four groups. The development of complex algorithms like deep learning or reinforcement learning, or the implementation of hybrid systems that combine many algorithms for more precise suggestions are examples of ways to improve existing systems.

The fundamentals of how music recommendation systems operate are described in the next section, ‘Music Recommendation Systems’. The use of music recommendation systems has grown in popularity in recent years, leading to the development of numerous techniques. Deep learning algorithms improve accuracy and efficiency, and personalisation and user-centric design are essential. The music recommendation system developed by Kumari and Khodhanpur makes excellent recommendations using item-based collaborative filtering and cosine similarity. Only a small amount of research has examined the relationship between user qualities and suggestions, including individual-level characteristics and group-level variables like country of origin.

The third part of the section explains how music RS's have changed over time. Important occurrences like the launch of streaming services and the surge in music consumption have had an impact on how music recommendation systems have developed. Automated music recommendation is essential in the age of extensive internet usage. Systems for making personalised recommendations based on user preferences and feelings are now crucial for music platforms thanks to technological improvements. Systems for recommending music come in many different forms, such as acoustic similarity-based approaches, collaborative filtering, and content-based methods.

The challenges faced are explained in the following section. Systems for making music recommendations have trouble with context awareness. Future research will concentrate on building compelling methodologies for personalisation, privacy, and security improvements. Rao et al. investigate collaborative filtering and content-based approaches, whereas Liang et al. provide a triplet neural network for music recommendation. It is crucial to make improvements in automatic playlist creation, context-aware music recommendations, and suggestions for the music business.

The fifth section provides an explanation of self-organising maps. Unsupervised neural network models called Self-Organizing Maps (SOMs) are employed for large-scale data processing such as dimension reduction, grouping, and similarity discovery. By utilising pre-trained CNNs and including topology-related enhancements, they can be enhanced. SOMs can be used to detect outliers, sample data, and uncover hidden patterns. Traditional SOMs and continuous learning algorithms are outperformed by the Dendritic-Self-Organizing Map (DendSOM) algorithm, which was developed after studying actual neurons.

The implementation of Self-Organizing Maps (SOMs) in music recommendation systems is examined in the final section, which is comparable to the SOM model that will be created. SOMs are used in the model to build a strong recommendation system, and it train on audio aspects and genres to identify complex musical associations. The strategy of utilising SOMs to improve recommendation accuracy in the model to be constructed is similar to the method of combining SOMs with other strategies like neural networks that have been explored. Although not all of the aforementioned techniques may be used in the implementation, the fundamental idea of using SOMs to comprehend and suggest musical qualities is still valid. In essence, the SOM model that will be developed demonstrates how SOMs successfully produce music suggestions by understanding fundamental song associations, in line with the concepts discussed.

## 3. METHODOLOGY

### 3.1. Data Collection

While searching for the appropriate dataset for this dissertation, it could be seen that there was no proper dataset that had audio features of the K-pop songs. To handle this, data was collected using Python and the Spotify API named Spotipy.

The steps involved in collecting the data are split into two parts.

1. Creating the Spotify developer's account.
2. Using the Spotipy API to get the data.

**Creating the Spotify Developers Account:**

A Spotify Developers Account is first created by navigating to the Spotify Developer Dashboard (https://developer.spotify.com/dashboard/) and creating a new account or logging

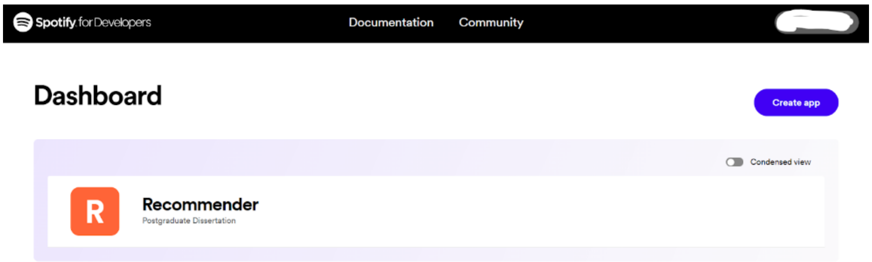


*Figure 1: Spotify Developer's account-Dashboard*

Once logged in, a new application is created to obtain the necessary API credentials (client ID and client secret). Give the application a name and description.

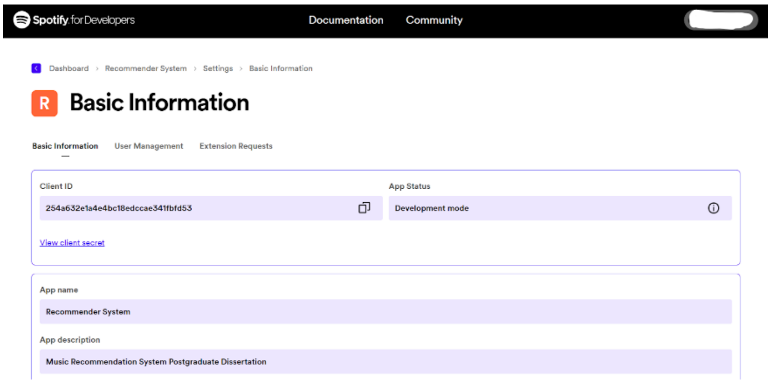


*Figure 2: Spotify Developer's account-Client ID and secret*

**Using the Spotipy API to get the data**

- The Python module 'Spotipy' is used to interact with the Spotify web API. Client ID and secret are provided for secure access.
- A list of playlist URLs is provided, representing 60 different genres and official playlists from individuals and groups.
- Each playlist is analyzed, and retrieved with details like song name, artist, genre, release date, and audio attributes like danceability, energy, and tempo. These features

provide an overview of a song's musical properties [49]. Both well-known and less well-known musicians are represented in the playlists chosen.

- Audio features are aspects of a song's sound. For example, "danceability" refers to how ideal a song is for dancing, "energy" refers to how energetic a song feels, and "tempo" refers to the music's speed. These audio features are retrieved for each song to help in providing an overview of a song's musical properties [66].

### 3.1.1. About the dataset

The dataset selected for this particular problem is the K-pop dataset. It contains categorical features such as genre and mode, numerical features like release date, danceability, energy, key, loudness, speechiness, acousticness, instrumentalness, liveness, valence, tempo, duration, time signature, and textual features like song name and artist.

The features extracted can be described as follows [42]:

| **AUDIO FEATURES** | **DATA TYPE** | **DESCRIPTION** |
|---|---|---|
| Genre | string | |
| Mode | integer | A track's tone (major or minor) is indicated by its mode.<br>Major is denoted by 1<br>Minor is denoted by 0 |
| Release Date | Datetime64 | The date when the song was released (DD-MM-YYYY format) |
| Danceability | float | Danceability ranges from 0.0 to 1.0, with 1.0 being the most danceable. It is based on a variety of musical features that show how apt a track is for dancing. |
| Energy | float | A subjective metric of intensity and activity, energy ranges from 0.0 to 1.0. Typically, intense music feels loud, fast, and noisy |
| Key | integer | This indicates the overall pitch/key of the song |
| Loudness | float | This indicates how loud the song is overall. |
| Speechiness | float | Speechiness is a feature that recognises spoken words in music.<br>Tracks with spoken words entirely: >0.66<br>Tracks with both music and speech. E.g.: rap (as parts/layers): 0.33 to 0.66<br>Tracks with no speech: <0.33 |
| Acousticness | float | This indicates whether the track is acoustic is indicated by a confidence value between 0.0 and 1.0. High trust in the track's acoustic nature is represented by a score of 1. |
| Instrumentalness | float | This determines if a track is vocal-free. Sounds like "ooh" and "aah" are seen as instrumental in this situation. |
| Liveness | float | This identifies whether there is an audience in the recording. Greater liveness values |

| | | |
|---|---|---|
| | | indicate a greater likelihood that the song was performed live. |
| Valence | float | This describes the overall musical positivity of the song and ranges from 0.0 to 1.0<br>High Valence indicates positivity (euphoric)<br>Low Valence indicates negativity (sadness) |
| Tempo | float | The estimated beats per minute (BPM) tempo of a track as a whole. It indicates the song's pace. |
| Duration | integer | Song's playtime in milliseconds |
| Time Signature | integer | A notation that is used to indicate how many beats there are in each bar.(e.g.: ¾) |
| Song Name | string | The title of the song |
| Artist | string | The main artist (and featured artist) of the song |
| Track ID | String | The track id of every song. This is a unique identifier |

*Table 1: Description of audio features*

### 3.1.2. Exploratory Data Analysis

In this section, we analyse the data and understand how the data is spread.

**Songs per Genre**

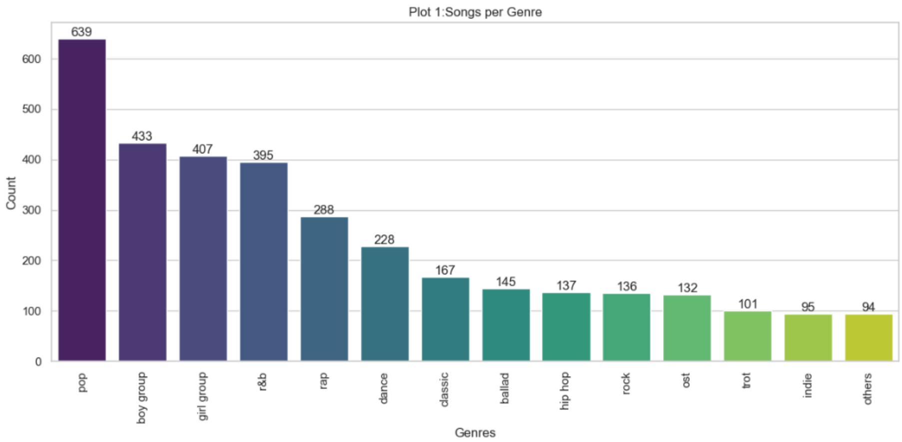


*Figure 3: Plot of Songs per Genre*

- In the above plot, it can be noted that, in the dataset, most of the songs are of the genre ‘Pop’ with a count of 639 songs.
- ‘Boy group’, ‘Girl group’, and ‘R&B’ have a comparable number of songs at 433, 407 and 395 songs respectively.
- Here, the genre ‘others’ indicates songs that did not have specific genres mentioned and there are around 94 songs indicated as ‘others’.

## Songs per Top 15 Artists

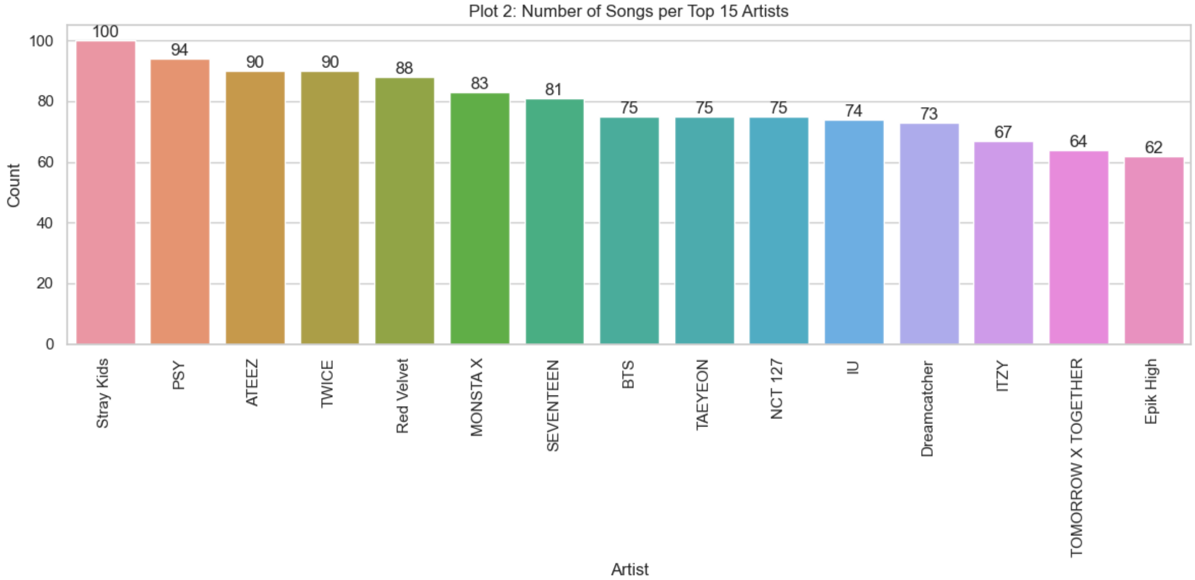


*Figure 4: Plot of Songs per top 15 artists*

- The above plot shows the number of songs released by the top 15 artists.
- 'Stray Kids' has the maximum number of songs and has released around 100 songs as a group and is ranked 1st while 'Epik High' has released 62 songs as a group and is ranked at the 15th position.
- 'Psy', 'Ateez', 'Twice. 'Red Velvet', 'Monsta X', and 'Seventeen' have released a comparable number of songs at 90,90,88,83 and 81 songs respectively.

## Year-wise Song Count

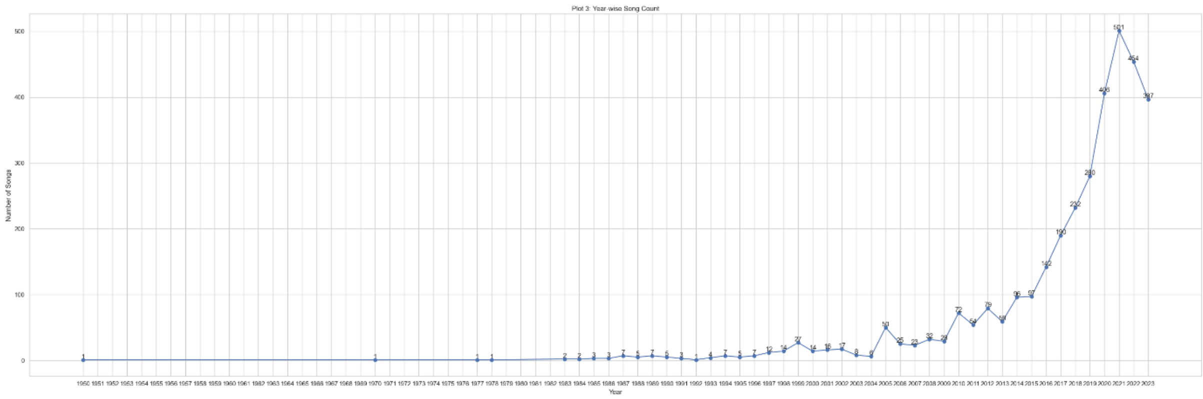

*Figure 5: Plot of Year-wise Song count*

From the above plot, it can be noted that not many songs were released from the 50's to the 90's. After the turn of the century, the release of new songs gradually increased with a peak in the year 2021 when 501 songs were released. Currently, in the year 2023, there has only been a release of 397 songs and this can be because the year is not yet over and there are chances that more artists will release songs.

## Valence, Tempo and Liveness- Distribution

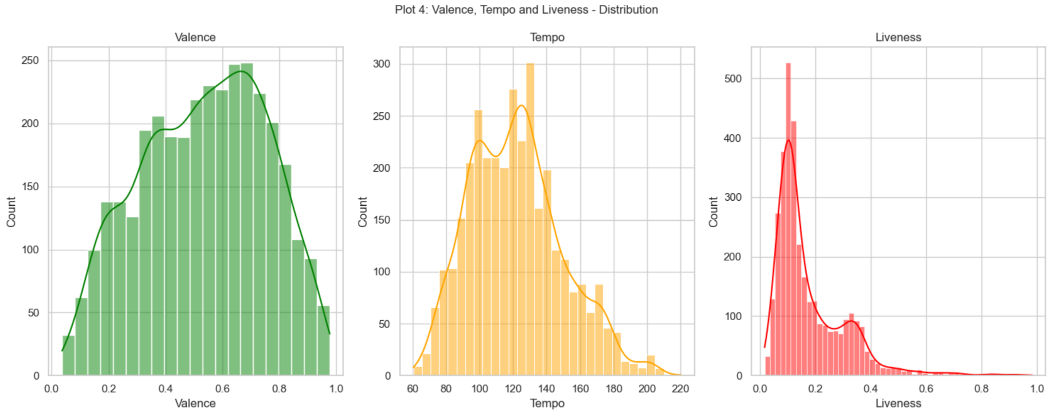


*Figure 6: Plot of the distribution of valence, tempo and liveness*

**Valence**: Most of the data points are towards 1. This shows that most of the songs are of a positive nature.
**Tempo**: The data is distributed in a way where most of the songs have a tempo ranging from 100 to 140.
**Liveness**: The data points are towards 0 which denotes that the songs in the data were recorded in a studio and not performed live

## Songs per Genre and Mode

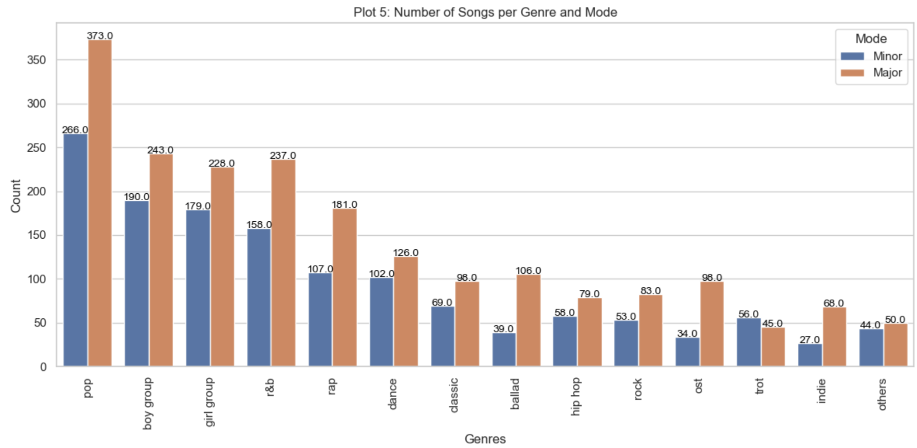


*Figure 7:Plot of Songs per Genre and Mode*

At first glance, it can be noted that songs across all the genres have an affinity towards being Major in mode i.e. they sound bright and cheerful rather than being minor i.e. they do not sound sad.

## Trends over the decades

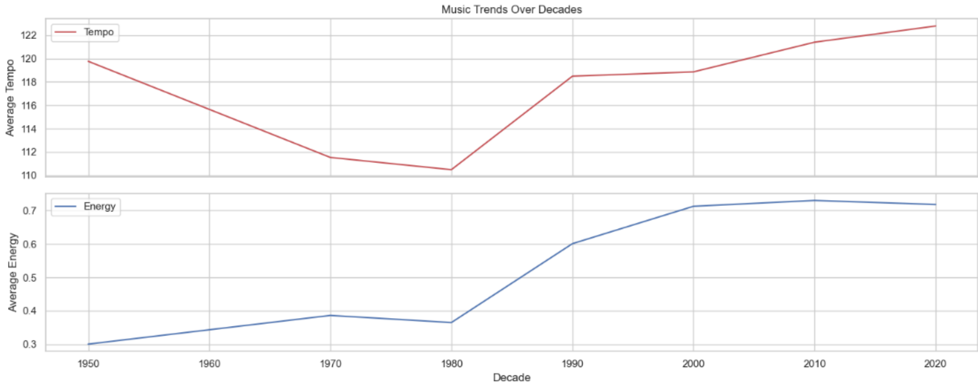


*Figure 8:Plot of Trends over the decades*

- It can be noted that through the 50s-70's, the tempo decreases and the energy increases.
- But, from the 80's both are gradually increasing simultaneously while reaching a peak in 2020.
- This might be a result of how K-pop has developed, technological developments in the music industry, or even shifting global music trends.

### 3.2. Self-Organising Maps

Due to recent technological advancements, artificial intelligence (AI) systems known as neural networks have grown in popularity [1]. Input, hidden, and output node layers make up artificial neural networks (ANNs), which rely on training data to become more accurate over time. Each node has a weight and threshold, and if its output exceeds the threshold, it activates and passes data to the next layer. [2]

Neural networks come in a variety of forms, and each has a special application. One such variation of the neural network, commonly known as Kohonen's map, is the self-organising map (SOM) [1].

#### 3.2.1. What is a self-organising map?

A self-organising map, or SOM for short, is an unsupervised neural network developed by Kohonen that reduces data dimensionality while maintaining topological qualities in the input space by using competitive learning strategies to build a low-dimensional representation of training samples [1]. SOM is used for mapping and clustering (or dimensionality reduction) processes to map multidimensional data onto lower-dimensional spaces to simplify complex situations for easy comprehension. The SOM is made up of two layers: the input layer and the output layer. They are also called Kohonen Maps, named after the creator [3].

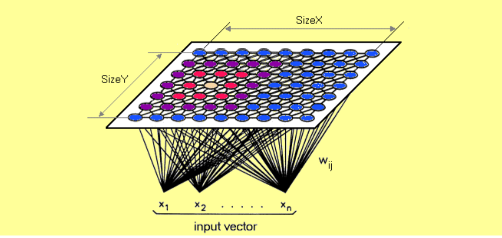


*Figure 9: Architecture of SOM [4]*

To make it simpler,

**"Self-Organising"**: Instead of focusing on error reduction, SOMs use competitive learning, which forces the network's neurons to compete with one another for activation based on the input data. Using distance-based metrics between data points and the learned representation, the feature vectors—the input data—are mapped to lower-dimensional representations. The learned representation is "self-organised" because it does not require any additional processing [44].

**"Map"**: A U-matrix, a map that depicts the separation between the neurons, serves as the representation for the weights learned by SOMs. Neighbour neurons in the SOM will be activated by inputs that are similar to one another. As a result, elements of the final "map" that include similar inputs will be grouped together [44].

The initialization of the network with weights, the selection of the Best Matching Unit (BMU) for each training data point, and the iterative updating of the neighbouring nodes' weights until all training data has been processed are the initial phases in training a network using a self-organising map, where data points compete for representation [45].

SOMs are a popular choice for many applications because they can handle a variety of tasks, making them a powerful and versatile tool for unsupervised learning. They are used for the following reasons [43].

- Data visualisation: to express high-dimensional data in a low-dimensional setting
- Anomaly detection: to find data points that significantly differ from the average or predicted behaviour
- Feature extraction is used to determine the most significant features or trends in a da-taset.
- Clustering: Based on proximity in the feature space, similar data points are grouped together.

### 3.2.2. <u>How do SOMs work?</u>

- Two crucial layers comprise self-organising maps: the input and output layers, com-monly called feature maps. The input layer is the initial layer in a self-organising map.
- The initialisation of the weight to vectors starts the mapping processes of the self-or-ganising maps.
- The mapped vectors are then examined to determine which weight most accurately represents the chosen sample, using a random vector as the sample.
- Neighbouring weights that are near each weighted vector are present.
- The chosen weight is then rewarded by having the option to turn it into a vector for a random sample.

- This encourages the map to develop and take on new forms. In a 2D feature space, they typically form square or hexagonal shapes.
- More than 1,000 iterations are spent repeatedly performing this entire process. [1]

## 3.3. Working

To build a SOM-based recommender system, the following steps are followed.

1. Data Loading and preprocessing
2. Feature Extraction
3. Model Building
4. Getting recommendations

The recommendations are obtained using Cosine similarity. Cosine similarity is a statistic for determining how similar two vectors are. The cosine of the angle between two vectors is used to calculate their similarity.[60])

### 3.3.1. Algorithm of the model

**Algorithm**: Music Recommendation System using Self-Organising Map (SOM)

**Input**:

⇒ A dataset containing song information (e.g., song names, artists, genres, and numerical features).

⇒ User input, including genre preference and numerical feature values (e.g., danceability, energy, etc.).

**Output**: Top N song recommendations based on user preferences.

**Steps:**

**Data Loading and Preprocessing:**

⇒ Load the song dataset.

⇒ Check for missing values

⇒ Remove duplicate entries based on song names.

⇒ Convert the "Release Date" column to a datetime format and extract the year.

**Model Building - Preliminary Steps:**

⇒ Select relevant columns from the dataset

⇒ Encode the "Genres" column using LabelEncoder.

⇒ Standardize numerical features.

⇒ Prepare the data for building the recommendation model.

**Model Building - Self-Organizing Map (SOM):**

⇒ Define SOM parameters:

⇒ Grid size (rows and columns).

⇒ Input dimensions (number of features).

⇒ Learning rate and sigma values.

⇒ Initialize the SOM with random weights.

⇒ Train the SOM on the standardized data for a specified number of iterations.

**Recommendation Function (Recommender):**

⇒ **Input:** User genre preference and numerical feature values.

⇒ Filter the dataset to include only songs of the specified genre.

⇒ Standardize the user's numerical feature values.

⇒ Find the Best-Matching Unit (BMU) for the user's input using the trained SOM.

⇒ Calculate the cosine distance between the user's input BMU and each song in the filtered genre.

⇒ Rank songs by their cosine distance to the user's input BMU.

⇒ Return the top N song recommendations.

### 3.3.2. Data Loading and Preprocessing

**Loading the data:** The data that was collected is loaded here.

**Missing Values Check:** The number of missing values in each dataset column is checked. It was found that there were no missing values in the dataset.

```
Song Name           0
Artist              0
Genres              0
Release Date        0
Danceability        0
Energy              0
Key                 0
Loudness            0
Mode                0
Speechiness         0
Acousticness        0
Instrumentalness    0
Liveness            0
Valence             0
Tempo               0
Duration_ms         0
Time_signature      0
dtype: int64
```

*Figure 10: Checking for missing values(No missing values found)*

**Removing Duplicates:** The dataset is checked for duplicate values. Duplicate values are checked based on the column 'Song Name'. If duplicate values are present in the dataset, then the first occurrence is retained and the repeated values are removed.

```
Dataset Size: 4010
Number of duplicates: 1070
Duplicates removed. New dataset size: 3397
```

*Figure 11: Checking and removing duplicates*

**Date Conversion and Year Extraction:** The column "Release_Date" is first converted to datetime64[ns] data type. This is done so that the year can be extracted from the date. The extracted years are stored in a new column "Year" and will be used for building the model.

### 3.3.3. Model Building- Preliminary Steps

**Feature Selection:** A subset of columns is selected from the original dataset and a new data frame is created. The selected columns are

- Song Name
- Artist
- Genres
- Danceability
- Energy
- Acousticness
- Valence
- Tempo
- Track ID

**Label Encoding:** The column ‘genre’ consists of categorical values. Categorical values cannot be used for building a model. Hence ‘genre’ is encoded using a label encoder into numerical values so that the SOM algorithm can work with them. Each unique genre is assigned a unique integer value

**Feature Extraction:** Here, a subset of the columns from the new data frame is selected and used to create a feature matrix. This matrix contains all the features that will be used for model building. The selected features are the encoded "Genres," "Danceability," "Energy," "Acousticness," "Valence," and "Tempo."

**Standardisation:** The feature matrix that was selected is standardised. Standardization ensures that all the features have a mean of 0 and a standard deviation of 1. This is important for some algorithms, such as those based on distances or gradients, as it brings all features to the same scale.

### 3.3.4. Model Building

**Define SOM Parameters:** First, the SOM's parameters are defined. The variables *rows* and *cols* indicate how many rows and columns there are in the SOM grid. They determine the grid's dimensions, which in this instance are 20x20. The variable *dim* indicates how many dimensions are included in the input data and it is determinedbased on the number of columns in the standardised feature matrix.

**Initialise SOM:** Next, an instance of the MiniSom class is created using the provided parameters (from the minisom library). The parameters are *learning_rate*, *sigma*, *dim*, *rows*, and *cols*. The SOM grid's dimensions are shown by the *rows* and *cols* variables, while the *dim* variable shows how many input dimensions (features) there are.

The parameter *sigma* regulates how far neighbouring nodes impact clustering during training. The influence is greater when the value is higher. The *learning_rate* parameter determines how quickly the SOM adjusts to the input data during training. An increased value causes faster adaptability.

**Initialise weights:** The weights of the SOM's neurons are initialised randomly using the standardised feature matrix in the subsequent phase. This initialization process is frequently used in SOM training to reduce bias and promote discovery.

**Train the SOM model:** The SOM will be trained in this following stage using a random selection of input samples from the feature matrix. The number of iterations is set to 1000, although this can be changed based on testing and the desired level of convergence. The weights of the neurons are changed during training to fit the distribution of the input data.

### 3.3.5. Getting Recommendations

**Input from the User**: The user offers certain information, such as their preferred musical genre and a few numerical attributes that define the music they enjoy. These numbers could indicate how upbeat, energising, acoustic, joyful, or fast-paced the music ought to be.

**Filtering songs based on the genre**: The function first examines a large list of songs and eliminates those that do not fit the user's selected musical style. Therefore, if the user prefers "Ballad" music, it will only take songs from that genre into account.

**Standardising the User's Input:** The values provided by the user may be on varying scales, such as one that ranges from 0 to 1 and 1 to 10, etc. In order to correctly compare them, the function ensures that they are on the same scale.

**Finding the Best Matching Songs:** The function then searches through the genre-specific song collection for songs that most closely match the user's choices. It achieves this by selecting the music from the list that most closely resembles user preferences. The "Best-Matching Unit" or BMU is what is used for this.

**Song Distance Calculation:** The function determines how similar each song in the genre is to the user-provided preferences. This is accomplished using a metric known as "cosine similarity." In short, it determines how closely the song's qualities—such as danceability, energy, etc.—match the preferences.

**Sorting and Recommending Songs**: Songs are ranked from the most similar to the least similar after the similarity for each song has been determined. The songs that are closest to our musical preferences are suggested as the best options.

**Getting the Recommendations:** The function provides a list of songs that are the top recommendations based on the user’s tastes, along with the names and artists of the songs. The default is 12 recommendations, but the user is free to request for more.

To summarise, this function takes the user's musical preferences into account, looks for songs in the preferred genre that fit the user's likes, and then provides us with a list of suggested songs that are most like what the user will enjoy. It's like getting customised music recommendations based on user preferences

## 3.4. Building the user interface

Before the user interface can be built, we will first pickle the following

- The data frame
- The standard scaler
- The label encoder
- The SOM model

The Pickle module in Python is a common method for serialising and de-serialising data types. Instead of rerunning code every time, object serialisation allows a data structure to be stored in memory and loaded as needed without losing its present state [61].

### 3.4.1. Creating the front end

- Inside the "templates" folder of the Flask project create HTML files for each web page. For this application, we create two files namely index.html and recommendations.html
- The index.html file is a form that gets the user's input.
- The recommendaions.html is the page which displays the song recommendations as Spotify iFrames.

### 3.4.2. Creating the Script

- The user interface is going to be built with the help of Flask, a compact and lightweight Python web framework that offers practical features and tools to make building web applications easier.
- Before starting, the Flask has to be installed.
- Make a flask application and specify an instance for running it.
- Define the paths (URLs) that link to particular functions.
- For each route, construct Python functions. These functions respond to queries sent to certain URLs. The function defined here is the recommender function that was used before.
- To process the user's data and handle incoming requests, employ request methods (such as GET, POST, etc.). and use return statements to return responses.

### 3.4.3. Integrating the front end with the script

- Create a folder for templates, then render them using the render_template function. These will serve as the application's front end.
- To render the HTML template, use the Flask route function's render_template function.
- The template index.html is a form that receives user input. Methods such as request.form can be used in Flask routes to handle form submissions.
- The user inputs are provided to the script's recommender function, and the resulting songs are passed to the other HTML page 'recommendations.html'.
- Launch the program.

## 4. RESULTS

### 4.1. Visualising the SOM

In this part, we can see how to SOM model is trained. This can be achieved by using a type of visualisation tool called a unified distance matrix (also called a umatrix). A U-matrix is a graphical representation of the distances between neurons in the dimension space of the input data. Specifically, we use the training vector to calculate the distance between neighbouring neurons [67].

Below are two plots that represent the SOM before training and after training.

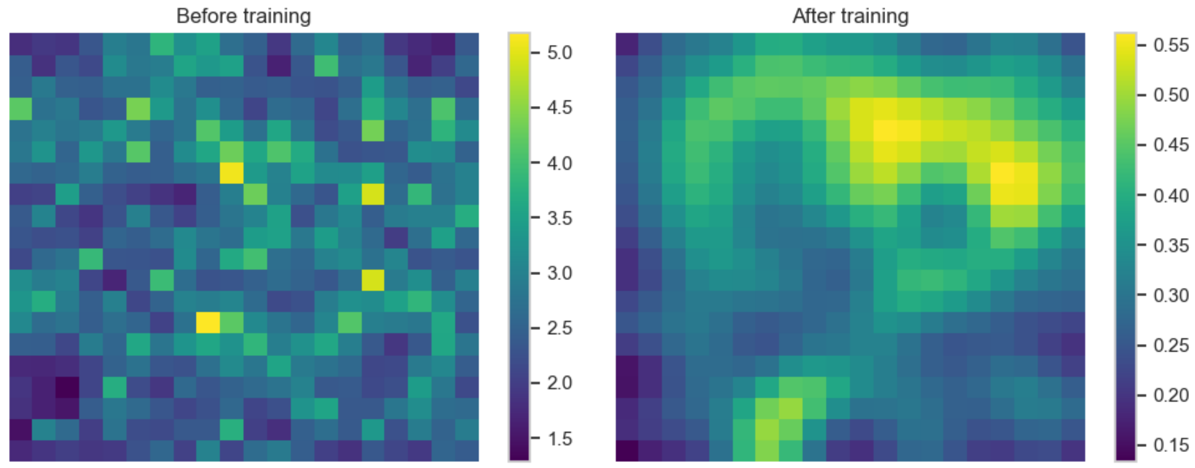


*Figure 12: Visualising SOM before and after training*

- The colour in the grid denotes the distance between adjacent neurons, and each cell in the grid represents a neuron (node) in the SOM.
- The areas that are darker are those where neurons are densely packed together and separated by less space, showing strong similarities.
- Greater distances between neurons are shown by lighter regions, which show differences or transitions between clusters.

When comparing the two plots, it can be seen that the SOM neurons have converged to represent clusters after training.

### 4.2. Testing the SOM

After the recommender function (user-defined function for getting recommendations) is created, we are going to test it now.

For this, the user gives the following as the input 'dance', 0.997,0.837,0.313,0.831,120 as input where 'dance' denotes the genre `and the number denotes the numerical features 'danceability', 'energy', 'acousticness', 'valence', and 'tempo.

These inputs are passed as parameters and the function is called and the following is given as the output by the model.

| | Index | Song Name | Artist |
|---|---|---|---|
| 0 | 173 | Clap Your Hands (박수쳐) | 2NE1 |
| 1 | 157 | 안녕 개비! (Sung by NMIXX) | NMIXX |
| 2 | 25 | ANTIFRAGILE | LE SSERAFIM |
| 3 | 159 | 안녕 개비! (Sung by NMIXX) (Inst.) | NMIXX |
| 4 | 5 | The Feels | TWICE |
| 5 | 190 | Up! | Kep1er |
| 6 | 196 | Mystery | Beast |
| 7 | 204 | SEXY LOVE | T-ARA |
| 8 | 201 | Come Back Again | INFINITE |
| 9 | 110 | Lights Out | SUNMI |
| 10 | 76 | Lemon (feat. Colde) | CHUNG HA |
| 11 | 18 | SG (with Ozuna, Megan Thee Stallion & LISA of ... | DJ Snake |

*Figure 13: Testing the model with user input that includes the genre 'dance' and other numerical values*

Next, we try with another set of user input values such as ‘ost, 0.665,0.848,0.0482,0.687,119.986 where ‘ost is the user’s desired genre. The following is as the output by the model.

| | Index | Song Name | Artist |
|---|---|---|---|
| 0 | 31 | Fall in Love | JEONG SEWOON |
| 1 | 95 | Like You Was To Me | Ha Hyun Sang |
| 2 | 3 | Yours | JIN |
| 3 | 97 | Love you again | MeloMance |
| 4 | 64 | SEE YOU SOON | HYUN SU |
| 5 | 108 | Pastel | Whee In |
| 6 | 33 | Secret | YELO |
| 7 | 74 | Susanne | Jukjae |
| 8 | 46 | Behind | Yang Da Il |
| 9 | 27 | We are still | Primary |
| 10 | 13 | The Painted On The Moonlight | MIYEON |
| 11 | 40 | Let's Stay Well (My love X Roy Kim) | Roy Kim |

*Figure 14: Testing the model with another set of user input that includes the genre 'ost' and other numerical attributes*

### 4.3. “MeloSOMmatic”- the SOM based recommender

“MeloSOMmatic” is the result of the above SOM. The word MeloSOMmatic is a combination of

“Melo”: ‘melody’ that emphasises on music

“SOM”: an acronym for self-organising maps

“matic”: the suffix implies a systematic and methodical approach

This platform will act as a medium (or user interface) and be a bridge between the user and the underlying SOM model, allowing users to provide input and receive recommendations.

Now, let us give the same inputs from above into our system.

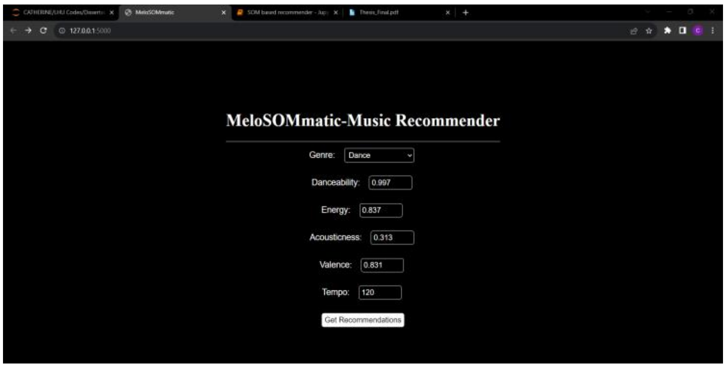


*Figure 15: Testing is done on the platform with the user inputs that were used in the above model for the genre 'dance'*

After the user inputs their preferance using the above form, we get the following as the result.

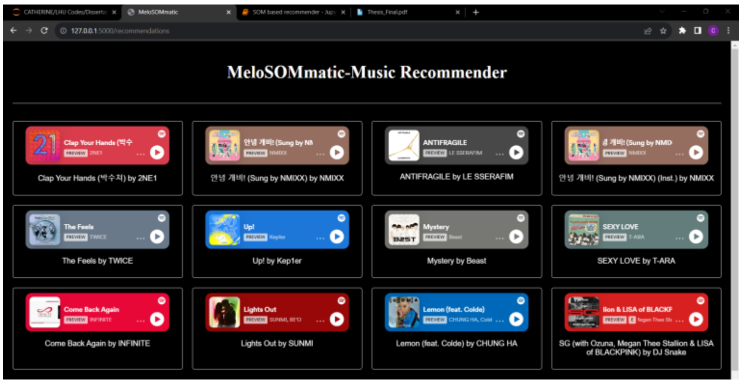


*Figure 16: The resulting output for the above inputs is displayed as Spotify iFrames*

It can be seen that we get the same songs we got from testing the model mentioned previously. The songs are displayed as Spotify iFrames.

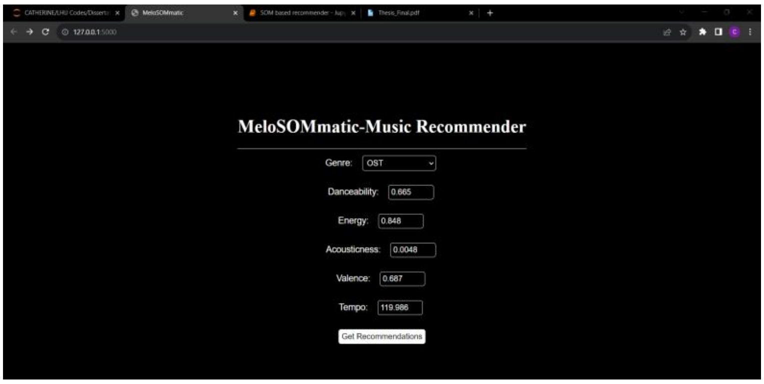


*Figure 17:Testing is done again on the platform with the user inputs that were used in the above model for the genre 'ost'*

We obtain the following result after the user enters their preference using the form mentioned above.

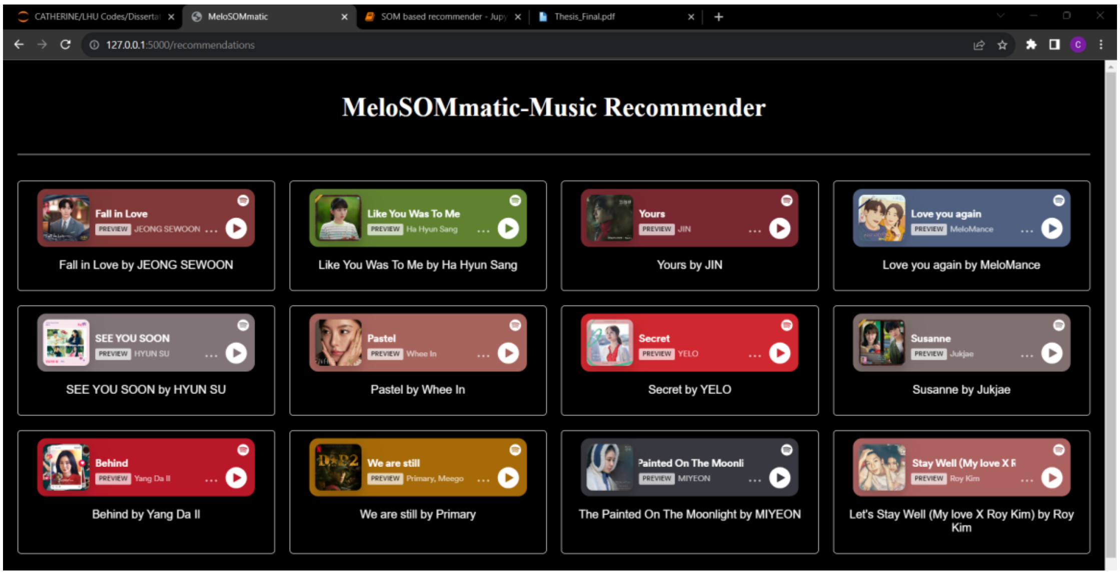


*Figure 18: The resulting output for the above inputs is displayed as Spotify iFrames*

It can be observed that we get the same songs as when we tested the previous model. Spotify iFrames are used to display the tracks.

These iFrames enable the platform to seamlessly integrate the ability to listen to a preview of the song, boosting the entire user experience. The use of Spotify dramatically increases the accessibility and variety of songs available to users, allowing them to discover and enjoy a wide selection of music right at their fingertips. Spotify iFrames are a practical and user-friendly feature that ensures users can listen to their favourite music while engaging with the platform's content.

It is clear that the above model works properly. When it is integrated with the platform created using the Flask server, the songs are displayed properly in an aesthetic manner.

## 5. LIMITATIONS

**Limited to Numerical features:** Although the previously mentioned Self-Organizing Map (SOM) technique has proven useful for dealing with numerical data, its inability to directly accommodate category or text-based elements poses a significant difficulty. This limitation has the possibility to have a significant impact on the use of vital information that influences listeners' tastes and experiences in the music industry. Lyrics, and artist biographies all play critical roles in shaping people's musical tastes and exposing them to new music. It is necessary to study alternative approaches or methods capable of addressing the gap left by the SOM's inability. This allows for the development of stronger music recommendation systems that respond to the various aspects that contribute to a satisfying and engaging experience for music enthusiasts.

**Scalability**: SOMs can become computationally expensive and difficult to train when working with large datasets or high-dimensional feature fields. We used a dataset of 4,000 records in our model, which may not appear to be a significant quantity. However, keeping in mind, that as the overall number of songs increases, so will the time required to train the model and generate recommendations. This possible issue could lead to poor performance and long wait times for users looking for music recommendations, limiting the model's practicality and real-world applicability.

**Cold start problem:** If some genres are underrepresented in the dataset, the SOM might not provide accurate recommendations for those categories. For instance, in our dataset, the genres ballad, hip hop, rock, ost, indie and trot have less than 200 records each when compared with the other genres like pop, boy group, girl group. So, when those genres are selected, because of the number of songs being less, the recommendations will not be. This is known as the "cold start" problem, where the system struggles to make recommendations for rarely seen items.

To overcome this issue and enhance the overall effectiveness of music recommendations, increasing the volume and diversity of data is crucial. By collecting and incorporating more information about these lesser-known genres into the existing dataset, the SOM's reliability in providing accurate recommendations can be substantially improved, thus mitigating the impact of the "cold start" problem.

**Grid Size**: The step of defining the size of a Self-Organizing Map (SOM) grid is critical since it can considerably influence the model's effectiveness and quality of recommendations. It is critical to predefine this grid size before training because any changes made later will require not only minor changes but perhaps a complete retraining of the model. As a result, significant money and effort may be expended. Choosing an ideal grid size is difficult due to the multiple elements that must be considered when defining the ideal setup for a specific use case.

## 6. FUTURE WORK

The Self-Organizing Map (SOM) based music recommender system that has been deployed using Flask is a promising platform for music aficionados to discover new songs and artists based on their tastes. While the current implementation gives helpful suggestions, there are various future improvements and enhancements that could further enhance the system's effectiveness and user experience.

Here are some possible future improvements for this SOM-based music recommender:

**Data Augmentation:** The diversity of the recommendations can be increased by expanding the dataset with new songs, genres and subgenres. While expanding, it is also essential to take into account various and underappreciated artists. This strategy not only broadens the recommendations but also makes more people aware of the underappreciated artists. The data grows more extensive and inclusive by including a variety of artists, providing users with a larger selection of music to explore.

**Integration of audio data**: It would be beneficial to incorporate audio data and extract features like rhythm, pitch, and harmony in order to deliver recommendations that are more informed musically. Users will be exposed to songs that share similarities in rhythm, pitch, and harmony in addition to songs that are similar in genre or artist by including these aspects in the recommendation system. This would enable a more comprehensive understanding of the user's musical interests and result in more precise recommendations.

**Accessibility:** It is important to include features that cater to users with disabilities' particular needs in order to make the recommender system accessible to them. This involves adding voice commands, allowing users to communicate with the system verbally, and incorporating screen readers to improve accessibility for users who are visually challenged.

**User feedback:** To enhance the recommendation algorithms, a feedback system that allows users to score and comment on suggested music would be a great addition. It gives the system the chance to better understand individual likes and modify recommendations by allowing users to share their preferences and thoughts.

## 7. CONCLUSION

In this dissertation, we embark on a venture exploring the use of Self-Organizing Maps (SOM) in the field of music recommendation, with a special emphasis on the vibrant and varied world of K-pop. Our exploration into this intriguing topic was driven by our research question, "How can I use SOM to create a music recommender platform for the field of K-pop?" Results from the journey are encouraging. By taking user tastes and needs into account, our SOM-based model successfully recommends fresh K-pop music. The “MeloSOMmatic” platform, which was developed as a result of our research efforts, not only exemplifies SOM's skills but also its dedication to improving the music discovery process.

But it's important to acknowledge certain limitations. These include dependence on limited numerical features, the difficulty in addressing the cold start issue for new users, problems regarding the size of the SOM grid, and scaling issues for larger datasets.

Future research in this area has to be centred on data augmentation to enhance feature sets, audio data integration for more thorough music analysis, improving accessibility for a larger user base, and actively seeking and incorporating user feedback to continuously improve the recommender system.

In conclusion, this study not only advances the field of music recommendation but also paves the way for future developments in SOM-based platforms for tailored content curation, particularly in the dynamic world of K-pop. The " MeloSOMmatic " initiative offers a useful starting point for upcoming advancements and novel ideas in the field of music recommendation systems.

## 8. REFERENCES AND BIBLIOGRAPHY


[1] V. Dey, “Beginners Guide to self-organizing maps,” Analytics India Magazine, https://analyticsindiamag.com/beginners-guide-to-self-organizing-maps/ (accessed Aug. 14, 2023).

[2] “What are neural networks?” IBM, https://www.ibm.com/topics/neuralnetworks#:~:text=Neural%20networks%2C%20also%20known%20as,neurons%20signal%20to%20one%20another. (accessed Aug. 10, 2023).

[3] Simplilearn, “What are self organizing maps: Beginner’s guide to kohonen map: Simplilearn,” Simplilearn.com, https://www.simplilearn.com/self-organizing-kohonen-maps-article (accessed Aug. 5, 2023).

[4] A. KHAZRI, “Self organizing maps,” Medium, https://towardsdatascience.com/self-organizing-maps-1b7d2a84e065 (accessed Jul. 30, 2023).

[5] M. Cottrell, M. Olteanu, F. Rossi, and N. Villa-Vialaneix, “Self-organizing maps, theory and applications,” *REVISTA INVESTIGACI ́ON OPERACIONAL*, vol. 39, no. 1, 2018.

[6] A. Ali, “Self Organizing Map(SOM) with Practical Implementation,” Medium, https://medium.com/machine-learning-researcher/self-organizing-map-som-c296561e2117 (accessed Aug. 1, 2023).

[7] M. Kleć, in *Recommender Systems for Medicine and Music*, vol. 946, A. Wieczorkowska , Ed. Springer, Cham

[8] L. Porcaro, C. Castillo, and E. Gómez, “Diversity by design in Music Recommender Systems,” *Transactions of the International Society for Music Information Retrieval*, vol. 4, no. 1, pp. 114–126, 2021. doi:10.5334/tismir.106

[9] S. Kumari and B. I. Khodhanpur, “Music Recommendation and Discovery,” *INTERNATIONAL JOURNAL OF ENGINEERING DEVELOPMENT AND RESEARCH*, vol. 2, no. 2, Jun. 2014.

[10] Y. Deldjoo, M. Schedl, and P. Knees, *Content-driven Music Recommendation: Evolution, State of the Art, and Challenges*, vol. 2, Jul. 2021.

[11] K. Bhareti, S. Perera, S. Jamal, M. H. Pallege, V. Akash and S. Wiieweera, "A Literature Review of Recommendation Systems," *2020 IEEE International Conference for Innovation in Technology (INOCON)*, Bengaluru, India, 2020, pp. 1-7, doi: 10.1109/INOCON50539.2020.9298450.

[12] A. Fanca, A. Puscasiu, D. -I. Gota and H. Valean, "Recommendation Systems with Machine Learning," *2020 21th International Carpathian Control Conference (ICCC)*, High Tatras, Slovakia, 2020, pp. 1-6, doi: 10.1109/ICCC49264.2020.9257290.

[13] F. Fouss, “Introduction to recommender systems” 2010 doi: 10.4018/978-1-61520-841-8.CH003

[14] L. Dall’Acqua and I. M. Gironacci, “Literature Review of Recommendation Systems,” in *Transdisciplinary Perspectives on Risk Management and cyber intelligence*,

Hershey, PA: Information Science Reference, an imprint of IGI Global, 2021, pp. 119–120

[15] A. Nawrocka, A. Kot and M. Nawrocki, "Application of machine learning in recommendation systems," 2018 19th International Carpathian Control Conference (ICCC), Szilvasvarad, Hungary, 2018, pp. 328-331, doi: 10.1109/CarpathianCC.2018.8399650.

[16] A. Rao, B. Adarsh, and S. Tripathi, “Supervised Feature Learning for Music Recommendation,” in *Advances in Signal Processing and Intelligent Recognition Systems* , A. R. Aditya, Ed. 2020

[17] H. Han, X. Luo, T. Yang and Y. Shi, "Music Recommendation Based on Feature Similarity," 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), Chongqing, China, 2018, pp. 650-654, doi: 10.1109/IICSPI.2018.8690510.

[18] M. Schedl, P. Knees, and F. Gouyon, “New paths in music recommender systems research,” *Proceedings of the Eleventh ACM Conference on Recommender Systems*, 2017. doi:10.1145/3109859.3109934

[19] P. Deshmukh and G. Kale, “Music and movie recommendation system,” *International Journal of Engineering Trends and Technology*, vol. 61, no. 3, pp. 178–181, 2018. doi:10.14445/22315381/ijett-v61p229

[20] A. Sternberg, N. Guy, and D. Portman, “Music recommendation for influencing physiological state”

[21] C. T. Min, “Contents Analysis Based Music Recommendation System”

[22] M. A. Sultan, “A study of self- organizing maps(som) neural network using MATLAB,” *Research on Humanities and Social Sciences*, 2020. doi:10.7176/rhss/10-6-01

[23] B. Wundervald, “Cluster-based quotas for fairness improvements in music recommendation systems,” *International Journal of Multimedia Information Retrieval*, vol. 10, no. 1, pp. 25–32, 2021. doi:10.1007/s13735-020-00203-0

[24] V. P. Sharma, A. S. Gaded, D. Chaudhary, S. Kumar and S. Sharma, "Emotion-Based Music Recommendation System," 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2021, pp. 1-5, doi: 10.1109/ICRITO51393.2021.9596276.

[25] D. Paul, “A Survey of Music Recommendation Systems with a Proposed Music Recommendation System,” in *Emerging Technology in Modelling and Graphics*, S. Kundu, Ed. 2020

[26] M. Schedl, B. McFee, D. Bogdanov, and M. Kaminskas, “Music Recommender Systems,” in *Recommender Systems Handbook*, P. Knees, Ed. 2015

[27] A. Seth, “Genre prediction for music recommendation using Machine Learning,” *EPRA International Journal of Research & Development (IJRD)*, pp. 206–210, 2020. doi:10.36713/epra4283

[28] S. Paul, S. Singh, and S. Rajbhoj, “Personalized music recommendation system,” *SSRN Electronic Journal*, 2021. doi:10.2139/ssrn.3772631

[29] B. M. Silva and N. Marques, *Ubiquitous self-organizing maps*, Aug. 2013.

[30] D. Celińska-Kopczyńska and E. Kopczyński, “Non-Euclidean self-organizing maps,” *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, Jan. 2022. doi:10.24963/ijcai.2022/269

[31] F. Giobergia and E. Baralis, “Fast self-organizing maps training,” *2019 IEEE International Conference on Big Data (Big Data)*, Dec. 2019. doi:10.1109/bigdata47090.2019.9006055

[32] L. Elend and O. Kramer, “Self-organizing maps with convolutional layers,” *Advances in Intelligent Systems and Computing*, pp. 23–32, 2019. doi:10.1007/978-3-030-19642-4_3

[33] M. Cottrell, M. Olteanu, F. Rossi, and N. N. Villa-Vialaneix, “Self-Organizing Maps, Theory and Applications,” *Revista de Investigacion Operacional*, vol. 39, no. 1, Apr. 2018.

[34] K. Pinitas, S. Chavlis, and P. Poirazi, “Dendritic Self-Organizing Maps for Continual Learning.,” *arXiv: Neural and Evolutionary Computing*, Oct. 2021.

[35] A. Gepperth and B. Pfülb, “A rigorous link between self-organizing maps and Gaussian mixture models,” *Artificial Neural Networks and Machine Learning – ICANN 2020*, pp. 863–872, Sep. 2020. doi:10.1007/978-3-030-61616-8_69

[36] C. Desai, S. Bhadra, and M. Parekh, “Music recommendation system using Python,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 7, pp. 699–707, 2023. doi:10.22214/ijraset.2023.54740

[37] K. Meghana, E. Sudeeksha, A. Somanth, and Y. Srinivasulu, “Content-based movie recommendation system,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 1, pp. 992–996, 2023. doi:10.22214/ijraset.2023.48533

[38] A. Niyazov, E. Mikhailova, and O. Egorova, “Content-based music recommendation system,” *2021 29th Conference of Open Innovations Association (FRUCT)*, 2021. doi:10.23919/fruct52173.2021.9435533

[39] M. Pulis and J. Bajada, “Siamese neural networks for content-based cold-start music recommendation.,” *Fifteenth ACM Conference on Recommender Systems*, 2021. doi:10.1145/3460231.3478847

[40] D. Sun, “Using factor decomposition machine learning method to music recommendation,” *Complexity*, vol. 2021, pp. 1–10, 2021. doi:10.1155/2021/9913727

[41] H. Liang, D. Zeng, Y. Yu, and K. Oyama, “Personalized Music Recommendation with Triplet Network,” *arXiv: Information Retrieval*, Aug. 2019.

[42] “Get tracks’ audio features,” Web API Reference | Spotify for Developers, https://developer.spotify.com/documentation/web-api/reference/get-several-audio-features (accessed Aug. 29, 2023).

[43] I. Bernardo, “Unsupervised learning series -exploring self-organizing maps,” Medium, https://medium.com/towards-data-science/unsupervised-learning-series-exploring-self-organizing-maps-fe2efde9f7a1 (accessed Aug. 2, 2023).

[44] H. Anjum, “Self-organizing maps (SOMS)-introduced, explained, and implemented,” Medium, https://medium.com/@akhtarhaniya/self-organizing-maps-soms-introduced-explained-and-implemented-7f2f96d2c16b (accessed Aug. 3, 2023).

[45] V. Christopher, “How does self organizing algorithm works?,” Medium, https://medium.com/analytics-vidhya/how-does-self-organizing-algorithm-works-f0664af9bf04 (accessed Aug. 3, 2023).

[46] W. Igbasi, “Exploring the evolution of music download:,” Beat, https://vocal.media/beat/exploring-the-evolution-of-music-download (accessed Aug. 31, 2023).

[47] V. Manning-Schaffel, “A musicologist explains the science behind your taste in Music,” NBCNews.com, https://www.nbcnews.com/better/lifestyle/musicologist-explains-science-behind-your-taste-music-ncna1018336 (accessed Aug. 31, 2023).

[48] F. Pavón, J. Vega, and S. D. Canto, “SOM and Feature Weights Based Method for Dimensionality Reduction in Large Gauss Linear Models,” in *Statistical Learning and Data Sciences*, Springer Nature, 2015

[49] “K-pop,” Spotify, https://open.spotify.com/genre/0JQ5DAqbMKFGvOw3O4nLAf (accessed Jul. 30, 2023).

[50] N. Zhou, J. Tian, and M. Li, “Online recommendation based on incremental-input self-organizing map,” *Electronic Commerce Research and Applications*, vol. 50, p. 101096, 2021. doi:10.1016/j.elerap.2021.101096

[51] C. Desai, S. Bhadra, and M. Parekh, “Music recommendation system using Python,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 7, pp. 699–707, 2023. doi:10.22214/ijraset.2023.54740

[52] N. H. Liu, C. Y. Chiang, and H. M. Hsu, “Eliminating cold-start problem of music recommendation through SOM based sampling,” *Applied Mechanics and Materials*, vol. 278–280, pp. 1119–1123, Jan. 2013. doi:10.4028/www.scientific.net/amm.278-280.1119

[53] N. J. Petrelli, “Music: The universal language,” Healio, https://www.healio.com/news/hematology-oncology/20220126/music-the-universal-language#:~:text=As%20the%20title%20of%20this,supports%20certain%20types%20of%20behavior.(accessed Sep. 11, 2023).

[54] D. Victor, “Music as a universal language of healing,” Harmony & Healing, https://www.harmonyandhealing.org/music-as-a-universal-language-of-healing/#:~:text=The%20power%20of%20music%20lies%20not%20only%20in%20its%20ability,space%20for%20exploration%20and%20creativity. (accessed Sep. 11, 2023).

[55] N. C. Mineri and A. R. Purnomo, “Hallyu wave di Indonesia: Penerimaan Khalayak TERHADAP brand ambassador idol K-pop di Indonesia,” *MASSIVE: Jurnal Ilmu Komunikasi*, vol. 3, no. 1, p. 54, 2023. doi:10.35842/massive.v3i1.79

[56] J. Arnaud, “A brief history of K-pop,” The Los Angeles Film School, https://www.lafilm.edu/blog/a-brief-history-of-kpop/(accessed Sep. 11, 2023).

[57] B. Vincent, “A brief, condensed history of K-pop,” Teen Vogue, https://www.teenvogue.com/story/brief-history-of-k-pop (accessed Sep. 11, 2023).

[58] W. Igbasi, “The Evolution of Music Players:,” Beat, https://vocal.media/beat/the-evolution-of-music-players (accessed Sep. 11, 2023).

[59] CalypsoRoom Editorial Team, “Why music streaming platforms are becoming popular among gen Z,” calypsoroom, https://www.calypsoroom.com/Why-Music-Streaming-Platforms-are-Becoming-Popular-Among-Gen-Z.html#:~:text=You%20can%20access%20millions%20of,anywhere%20with%20an%20internet%20connection.(accessed Sep. 11, 2023).

[60] F. Karabiber, “Cosine similarity,” Learn Data Science - Tutorials, Books, Courses, and More, https://www.learndatasci.com/glossary/cosine-similarity/ (accessed Sep. 11, 2023).

[61] N. Selvaraj, “Python pickle tutorial: Object serialization,” DataCamp, https://www.datacamp.com/tutorial/pickle-python-tutorial (accessed Sep. 11, 2023).

[62] M. Tingle, “Getting Started with Spotify’s API & Spotipy,” Medium, Oct. 03, 2019. https://medium.com/@maxtingle/getting-started-with-spotifys-api-spotipy-197c3dc6353b (accessed Aug. 11, 2023)

[63] C. Watts, “Extracting Song Data From the Spotify API Using Python,” Medium, Feb. 10, 2022. https://towardsdatascience.com/extracting-song-data-from-the-spotify-api-using-python-b1e79388d50 (accessed Aug. 11, 2023)

[64]“Spotify Music API - Data Extraction - Part1,” kaggle.com. https://www.kaggle.com/code/pavansanagapati/spotify-music-api-data-extraction-part1 (accessed Aug. 11, 2023).

[65] Spotify, “Spotify,” Spotify, 2023. https://open.spotify.com/

[66] “Get track’s audio features,” Web API Reference | Spotify for Developers, https://developer.spotify.com/documentation/web-api/reference/get-audio-features (accessed Aug. 12, 2023).

[67] E. Alhoniemi, J. Hollmén, O. Simula, and J. Vesanto, “Process monitoring and modeling using the self-organizing map,” *Integrated Computer-Aided Engineering*, vol. 6, no. 1, pp. 3–14, 1999. doi:10.3233/ica-1999-6102

[68] NeoNeo, “Working of labelencoder in sklearn,” Stack Overflow, https://stackoverflow.com/questions/41773751/working-of-labelencoder-in-sklearn (accessed Jul. 31, 2023).

[69] S. Mulani, “Using StandardScaler() function to standardize python data,” DigitalOcean, https://www.digitalocean.com/community/tutorials/standardscaler-function-in-python (accessed Jul. 31, 2023).

[70] M. Saeed, “Self-organizing maps: Theory and implementation in python with NumPy,” Stack Abuse, https://stackabuse.com/self-organizing-maps-theory-and-implementation-in-python-with-numpy/ (accessed Jul. 31, 2023).

[71] “Sklearn.metrics.pairwise.cosine_distances,” scikit, https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_distances.html (accessed Aug. 4, 2023).

[72] thebrownviking20, “Clustering images w/ neural network & bayesian opt,” Kaggle, https://www.kaggle.com/code/thebrownviking20/clustering-images-w-neural-network-bayesian-opt#Self-Organizing-Maps (accessed Sep. 6, 2023).

[73] I. AHANGAMA, “U matrix of SOM for Iris dataset,” Kaggle, https://www.kaggle.com/code/imanthaahangama/u-matrix-of-som-for-iris-dataset (accessed Sep. 7, 2023).

[74] Long Thai, “Need a specific example of U-Matrix in self organizing map,” Stack Overflow, https://stackoverflow.com/questions/6485699/need-a-specific-example-of-u-matrix-in-self-organizing-map (accessed Sep. 7, 2023).

[75] “Minisom on Pypi,” Libraries.io, https://libraries.io/pypi/MiniSom (accessed Aug. 1, 2023).

[76] Pankaj, “Python pickle example,” DigitalOcean, https://www.digitalocean.com/community/tutorials/python-pickle-example (accessed Sep. 10, 2023).

[77] A. Dyouri, “How to make a web application using flask in python 3,” DigitalOcean, https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3 (accessed Sep. 11, 2023).

[78] J. J. Kamminga, “Home Minimal One Page website with pop-open menu ,” CodePen, https://codepen.io/jj-kamminga/pen/ExxGadG (accessed Sep. 10, 2023).

[79] “Using the IFRAME API,” Using the iFrame API | Spotify for Developers, https://developer.spotify.com/documentation/embeds/tutorials/using-the-iframe-api (accessed Sep. 10, 2023).

[80] R. Cheng, “Build a song recommendation system using Streamlit and deploy on Heroku,” Medium, https://towardsdatascience.com/build-a-song-recommendation-system-using-streamlit-and-deploy-on-heroku-375a57ce5e85 (accessed Sep. 9, 2023).