

Emotion Based Music Recommendation System Using Machine Learning Algorithm

Mrs. Sumera Jabeen¹, K.Kiran², E.Ramprasad², M.Sravani², K.Prasanna²

¹Asst.Professor, Computer Science and Engineering, CMR Engineering College, Medchal, T.S, India,

²B. Tech, Computer Science and Engineering, CMR Engineering College, Medchal, T.S, India

Abstract: Music is widely used for mood and emotion regulation in our daily life. As a result, many research works on music information retrieval and affective human-computer interaction have been proposed to model the relationships between emotion and music. However, most of these works focus on applications in a context-sensitive recommendation that considers the listener's emotional state, but few results have been obtained in studying systems for inducing future emotional states. This paper proposes *Moodify*, a novel music recommendation system based on reinforcement learning (RL) capable of inducing emotions in the user to support the interaction process in several usage scenarios (e.g., games, movies, smart spaces). Given a *target* emotional state, and starting from the assumption that an emotional state is entirely determined by a sequence of recently played music tracks, the proposed RL method is designed to learn how to select the list of music pieces that better "match" the target emotional state. Differently from previous works in the literature, the system is conceived to induce an emotional state starting from a current emotion instead of capturing the current emotion and suggesting certain songs that are thought to be suitable for that mood. We have deployed *Moodify* as a prototype web application, named *MoodifyWeb*. Finally, we enrolled 40 people to experiment *MoodifyWeb*, employing one million music playlists from the Spotify platform. This preliminary evaluation study aimed to analyze *MoodifyWeb*'s effectiveness and overall user satisfaction. The results showed a highly rated user satisfaction, system responsiveness, and appropriateness of the recommendation (up to 4.30, 4.45, and 4.75 on a 5-point Likert, respectively) and that such recommendations were better than they thought before using *MoodifyWeb* (6.45 on a 7-point Likert).

Keywords: affective computing; musical emotion; emotional state; reinforcement learning; user evaluation

1. Introduction

Music is an important aspect of any human culture, being able to induce a range of intense and complex emotions both in musicians involved in composing pieces and individuals listening to them. The digital age involved sizeable changes in the economy, in the industrial and social spheres, with interesting advances and transformations also in the music field. With regard to the music streaming market, its size was valued at USD29.45 billion in 2021 and is expected to expand at a compound annual growth rate (CAGR) of 14.7% from 2022 to 2030 [1]. The penetration of music streaming platforms, as well as the ubiquity of smartphones, will boost the music market growth, accordingly. Furthermore, platforms that allow streaming services are gaining popularity, offering services such as song recommendations and automatic playlist personalizations by supporting individuals in suggesting similar (and preferred) pieces. On the other hand, emotion/mood has become a fundamental criterion used by digital technologies in predicting social behaviors or conditioning people in their social interactions and work activities.

In light of this, music systems that regulate mood and emotions in our daily life are arousing particular interest. Consequently, a great deal of research has been undertaken in the *affective computing* community to model the relationship between music and emotion [2–5]. More in general, applications of affective computing studies can be found in education, health care, entertainment, ambient intelligence, multimedia retrieval, and music retrieval and generation. As for the specific musical context, most of these works consist of context-sensitive recommendation tools that consider the listener's emotional state. Unfortunately, few results have been obtained in the study of music systems for *induction* of future emotional states, i.e., methods for influencing the emotional state of listeners and adapting interaction with technology to their affective state. Unlike games or movies [6,7], where the study of the induction of emotional states has recently obtained interesting results, especially in educational or commercial contexts, the potential of music as a means of inducing emotions still leaves significant possibilities for study. Inductive systems use *affective contents* for induction of emotional states, assuming that the emotions conveyed by the affective content (*perceived* emotions) are always consistent with

emotions brought to mind in users' (*induced* emotions) games or movies [2]. There are at least two main perspectives in such systems: user and system perspectives. Users perceive and interpret the content (perceived emotions). Systems usually provide *emotional annotations* that describe which emotions are expected by the users during an interaction step (intended emotions). In general, perceived and induced emotions are not usually considered separately in studies on affective content. However, some studies on "music emotions" have shown that in music, traditionally regarded as an art form that can make people produce emotional responses or induce their emotions naturally [8,9], emotions perceived are not always consistent with the emotions elicited in listeners [10].

This is particularly evident in most music recommendation systems. Recommender systems have been widely studied in recent years [11,12], but they do not always lead to the best possible designs for affective recommendation systems.

Several studies showed that emotions could play a significant role in designing intelligent music recommendation systems, and most of them focused on "recognizing" emotions induced by music [13–18], nearly no attempts have been made to model musical emotions and their changes over time in terms of a target "emotion to induce". In this paper, we explore this direction, and we focus on the problem of defining an intelligent music recommendation system that, given a future *target* emotional state to induce, and starting from the assumption that an emotional state is determined entirely by a sequence of music pieces recently listened to, selects the list of music pieces that better "match" it.

In order to "recommend" music for inducing a *target emotional state*, we exploit reinforcement learning (RL) techniques that were proven effective in recommendation music systems. The idea is to train an intelligent agent capable of recommending songs such that the user's mood changes from a given current emotional state to a desired target emotional state based on the user's musical preferences (see Figure 1 for an overview of the proposal). The agent learns the user's preferences and the best trajectories for inducing the target emotion through a feedback-based mechanism. However, to face sparse and deceptive problems, we propose a novel method based on Go-Explore [19]. Go-Explore has proven particularly effective on hard-exploration problems, a category that many real-world issues belong to. We will show that this also applies to the problem of affective computing in the musical recommendation.

The main contributions of this paper can be summarized as follows.

- We propose *Moodify*, a novel music recommendation system based on Go-Explore, which takes into account the listener's emotional state for inducing a future *target emotion*; the main novelty is that it adopts a "look-forward-recommendation", i.e., it recommends music intended to induce (in the future) a specific target emotion. Previous works in the literature have proposed methods that, based on the current user's mood, recommend music or artists to listen to, for example, by computing similarities between artists' and users' moods.
- To analyze its effectiveness and overall user satisfaction, we have involved 40 people in testing *Moodify*, with one million music playlists from the Spotify platform; results obtained show that the proposed system can bring both significant overall user satisfaction and high performance.
- The proposed method has been developed as a Web application, namely *MoodifyWeb*, which exploits Spotify API for developers and JavaScript. To the best of our knowledge, this is one of the first proposals deployed in software for end-user.

The remaining parts of the paper are structured as follows. First, in Section 2, we offer an overview of the most used recommendation approaches in the literature, and we discuss some relevant works that inspired the proposed one. With respect to these points, we place our proposal highlighting similarities and differences with previous works. Then, in Section 3, we provide preliminary knowledge to understand the methods and techniques used in *Moodify*. In Section 4, we formalize the music recommendation system highlighting the relation with Go-Explore. Next, Section 5 offers details on the web application implemented that revolves around the proposed music recommendation method. Section 6 is devoted to discussing the results obtained when surveying users about the usability and satisfaction of *MoodifyWeb*. Lastly, in Section 7, we provide final remarks with envisioned future directions of this research.

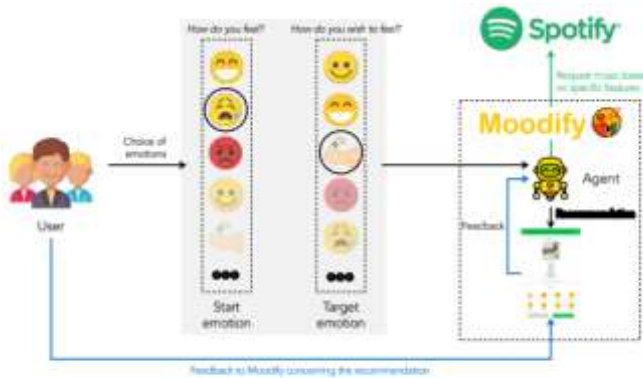


Figure 1. An overview of Moodify: the user selects the starting emotional state and the target emotional state. Such emotions are sent to the agent. The agent recommends a music track from Spotify, and expresses feedback through MoodifyWeb. Such feedback is, in turn, provided to the agent for refining the recommendation.

2. Related Work

In the literature, there are many solutions for recommending music suited to the listeners' environment, and in general, in all areas that refer to IT for "well-being", e.g., gyms [20] or home settings. From a *musical* point of view, the music generation takes place either by selecting existing music from platforms such as Spotify or Youtube or by using sophisticated generative music composition techniques. From a *technological* point of view, most such systems combine Artificial Intelligence (AI) and Internet of Things (IoT) techniques to ensure intelligent musical choices that satisfy listeners [21].

Therefore, given the vastness of the topic, in this section, we limit ourselves to an analysis of the works most closely related to the one proposed in this paper, essentially classifying them into four main categories: *collaborative filtering* (Section 2.1), *content-based filtering* (Section 2.2), *context-based filtering* (Section 2.3), and *emotion-based filtering* systems (Section 2.4).

2.1 Collaborative Filtering

Collaborative filtering generates automatic predictions about users' interests by collecting preferences from a large user base. This approach adopts *user ratings* to recommend songs. Such systems are built on the following assumption: users who rate songs similarly in the past will continue to rate them similarly in the future. Usually, clustering algorithms are employed to deliver recommendations. Ratings can be of two kinds: *explicit* or *implicit*. Examples of explicit ratings are the stars-based systems that e-commerce sites use: the user has to express a rating often based on a five-star score where the higher is the better (e.g., Trustpilot (Available online: <https://it.trustpilot.com>, accessed on 18 October 2022)). These ratings are explicitly provided by the users. Instead, implicit ratings can be collected by throwing light on the user's behaviors. For instance, play counts can be employed for implicit rating: a song played thousands of times gets a higher implicit rating than some others that have listened a dozen times. The biggest drawback of these kinds of systems is that they offer poor recommendations in the early stages. Especially for songs with very few ratings, recommendations are performed in a not-very-reliable fashion. This is a well-known issue in the literature, named the *cold-start* problem. When a new user joins the system, the recommender cannot offer effective suggestions, as the user has never interacted before and hence has not rated anything yet. Another challenge of collaborative filtering is related to human effort. In general, users are not willing to rate every item on a system that requires a lot of effort and attention to generate recommendations. Among the closest articles in the literature, there is the proposal by in which association rules and music features were added to a collaborative filtering mechanism. The system considers users' preferences for different song features and uses the similarity of interests among different users to suggest music. The system has been implemented in a Web application as we did, and the author also performed an experiment with 20 real users. The main difference is that we do not use a collaborative filtering method, but the suggestion is only tailored to the specific user; we employ an RL-based method to recommend music and not a rule-based algorithm; lastly, the deployed Web application is not described thoroughly as well as the user evaluation, which is, furthermore, only a preliminary one and does not involve, for example, *confirmation of expectation test*. In addition, the results of the evaluation are not clear.

Moodify employs an explicit rating “encapsulated” in an RL approach tailored to one user only. Every user has his/her own agent tailored, through usage, to his/her needs. Such a rating mechanism is used to define a reward function, i.e., by asking the user to evaluate how much the emotion felt at the end of each listening is similar to target emotion.

2.2. Content-Based Filtering

In the content-based filtering approach, music is recommended to exploit the system’s comparison between the items’ content and a user’s profile. Each item’s content is represented as a set of tags. In the case of textual documents, the tags can describe the words within a document. In the case of music, the tag—in the simplest form—can be related to the genre. Several issues must be considered when implementing such a category of systems:

- tags can either be assigned automatically or manually;
- the tags must be generated or assigned such that both the user’s profile and the items can be easily matched and compared to derive a similarity measure;
- a learning algorithm must be chosen that learns and classifies the user’s profile based on played songs (i.e., seen items) and offers recommendations based on it.

To recommend music, the song’s features, such as loudness, tempo, and danceability, are analyzed. Among the widely used methods to perform content-based filtering and measure similarities between user’s profile and songs are (i) clustering and (ii) expectation-maximization with Monte Carlo sampling. These techniques can recommend music tracks also with very little data; thus, they solve the cold-start problem (seen in Section 2.1). The major challenge of these approaches is in the appropriateness of the item. Another major drawback is that, with tags trying to describe the songs’ macro-characteristics, these approaches fail to differentiate crucial musical differences between similar songs in terms of tags.

In the authors introduce MoodPlay, a system for recommending music artists based on the general mood of the artists and the self-reported mood of users. The authors proposed the method and the visual (graph-based) interface of the system. In addition, they performed an experiment with more than 200 final users. From these experiments, it emerged that mood plays a crucial role in the recommendation. The main differences from this work are (i) we recommend songs, not artists; (ii) we only base our recommendations on the starting emotion and target emotion; thus, we do not consider artists’ general mood; (iii) our system is designed to induce a particular emotion, not to recommend a specific artist based on “the similarity” between certain user and artist moods; (iv) the ultimate goal of [20] was more related to understanding how users perceive recommendations through visual interfaces than generating an affective recommender system.

With respect to these kinds of systems, *Moodify* does not use pre-defined item content to compare with the user profile. It “dynamically” builds an intelligent agent capable of selecting the music most suited to the user’s target emotional state simply by observing the choices and the ratings assigned by the user himself during a training phase.

2.3 Context-Based Filtering

The context-based filtering approach takes advantage of the public perception of a music track in its suggestions. It exploits social media such as Facebook and Twitter and video platforms such as YouTube to collect information and derive insights about the public opinion of songs. Then, it recommends such music tracks accordingly to the users. This approach considers the users’ listening history of collecting user data; next, it recommends similar songs based on the engagement the songs have generated on social media. The context-based technique can build a “For You section” for the user through intelligent exploitation of user preferences (i.e., the listening history) and social media engagement of different music tracks. Another technique in this category of filtering uses the user’s location to suggest appropriate music tracks. The basic idea is that listeners in the same place may like similar music, and the system suggests music tracks with this assumption. The literature offered insights into the performance of this model, that is, it could perform as well as the amount of social

information collected but it needs to integrate with various sources and exploit a joint analysis of a massive data load to ensure good performance.

A different kind of context-based technique exploits data captured from the users, for example, from their activities that are treated as context. In the authors propose a smartphone-based mobile system to recognize human activities and recommend music accordingly. In the proposed method, a deep recurrent neural network is applied to obtain a high level of activity recognition accuracy from accelerometer signals on the smartphone. Music recommendation is performed using the relationship between recognized human activities and the music files that reflect user preference models in order to achieve high user satisfaction. The results of comprehensive experiments with real data confirm the accuracy of the proposed activity-aware music recommendation framework. In this case, the authors have not developed the system as an application for end-users, and they have not evaluated their method with listeners. Conversely, in the present work, we provide insights from end-users on the MoodifyWeb app deployed. Similarly, in [21], the author proposed a framework based on deep learning and IoT architectures to build a music recommendation system, but did not provide any software or evaluation to listeners. Both the aforementioned works revolve around the recognition of emotion through different devices and the recommendation of a suitable song. Differently, we aim to induce emotion through a series of songs with *Moodify*.

With respect to context-based filtering, our solution does not build a listening history nor collect information to be used for the recommendation. Instead, the listening history is implicitly employed in the training phase to build the agent and the reward of our method. The only listening information exploited concerns the audio features from Spotify of the songs listened to during the training sessions.

2.4 Emotion-Based Filtering

As explained above, music and human emotions are closely intertwined, so we have a recommended approach that considers human emotions, namely emotion-based filtering. Different audio features of the music tracks are used to understand emotions that they may trigger or induce. Then, music streaming sites build playlists based on human emotions and moods tailored to a feeling that a user might experience while listening to those songs. In this field, the research on affective computing has produced a series of interesting solutions (see for a recent survey on the topic related to music). We have identified various works. Some studies identify emotions through facial expressions. Others analyzed EEG physiological, and video signals. These works show that musical recommendation is generally carried out by combining physiological signals, heart and respiratory rates, and facial expressions, and in general, AI methods (generally deep learning techniques) were used to analyze such information. Among the works on this kind of filtering, we found where the authors propose an emotion-based music recommendation framework that learns the emotion of a user from the signals obtained via wearable sensors. In particular, a user's emotion is classified by a wearable computing device integrated with a galvanic skin response and photoplethysmography sensors. The experimental results were obtained from 32 subjects' data. The authors evaluated several machine learning methods, such as decision tree, support vector machines, and k-nearest neighbors. The results of experiments on real data confirmed the accuracy of the proposed recommender system. With respect to we deploy an RL-based recommender system for music to induce emotions in a prototype Web application, and we perform a real-world experiment with end-users to get their perceptions about *Moodify*.

Moodify belongs to this class. However, some novelties need to be highlighted. First, the equipment needed for the recommendation. Such solutions require EEG or ECG, facial expression, or physiological information to recommend adequate songs. However, the devices need to capture those traits for the mood analysis are not common and quite expensive in some cases. Our idea is that *Moodify* can recommend music without requiring further

devices or equipment. Though, *Moodify* can be extended with appropriate modules to consider traits like facial expression, and EEG for recognizing the mood while in use. Furthermore, *Moodify* adopts a "look-

forward-recommendation”, i.e., it recommends music with the aim of inducing (in the future) a specific target emotion. All the methods described, instead, adopt a “look-back-recommendation”, i.e., to recommend music only by using previously collected or observed information.

2.5 Summarizing Literature’s Proposals

In this section, we summarize the proposals available in the literature and we list the similarities and differences with ours. Such information is reported in Table 1, where we sketch the papers based on: (i) type of approach (collaborative, emotion, etc.); (ii) the idea behind the proposal; (iii) whether a method is presented; (iv) whether the software is presented/available; (v) whether a user evaluation/study has been carried out.

Table 1. Main points of both closest articles in the literature and this work. † = lacking details. * = framework.

Ref.	Approach	Idea	Method	Software	User Evaluation
[20]	Collaborative	Suggesting music based on similarities between users’ preferences and music features	✓	✓†	✓†
[21]	Content + Emotion	Suggesting music artists suitable to a specific mood based on similarities between artists’ and users’ moods	✓	✓	✓
[31]	Context	Capturing human activities via smartphones’ accelerometer and suggesting suitable music	✓	✗	✗
[21]	Context	Capturing human activities via IoT devices and suggesting suitable music	✓*	✗	✗
[6]	Emotion	Capturing emotions through wearable sensors suggesting music suitable to those emotions	✓	✗	✗
This work	Emotion	Induce emotion creating a trajectory of music songs to listen so to get an indicated target emotion given a starting emotion	✓	✓	✓

3. Background

This section provides some basic notions necessary to understand the proposed system. We first describe the model of emotions used in this work (Section 3.1). Next, we detail the audio features provided by Spotify (Section 3.2) and the RL-based method used to define the proposed music recommendation system (Sections 3.3 and 3.4).

3.1 Models of Emotional States

Emotions are biologically based reactions essential in determining behavior. Among the several models of emotions proposed in the literature, one of the most used is the *circumplex model* defined by Russell. Such a model organizes the emotional states in terms of *valence* and *arousal*. The result is a two-dimensional space, where a pleasant- unpleasant (valence) value is represented by the horizontal axis and high-low arousal is represented by the vertical axis (see Figure 2). As proposed in [6], in this work, we have used such a model by considering emotional states organized in the following groups: *pleasant-high* (excited, amused, happy), *pleasant-low* (glad, relaxed, calm), *unpleasant-high* (tired, bored, depressed), and *unpleasant-low* (frustrated, angry, tense). We remark that in this work, we are not interested in the recognition of emotional states. As we will see in Section 4, to build the Go-Explore-based model used by the proposed recommendation system, we have used the “user feedback” regarding the emotions induced by the musical pieces used during the training phase of the model itself.

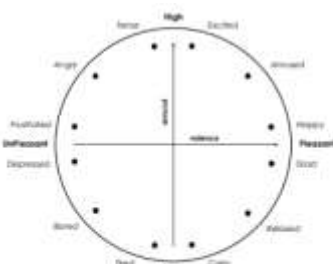


Figure 2. Examples of emotional states in the circumplex model. In this study, these states are divided into four groups: Pleasant-high, Pleasant-low, Unpleasant-high, and Unpleasant-low.

3.2. Spotify

Spotify Technology S.A. (Available online: <https://www.spotify.com/it/>, accessed on 18 October 2022) is a media-services provider whose core business is an audio streaming platform that provides access to over 50 million music tracks (Available online: <https://newsroom.spotify.com/company-info/>, accessed on 18 October 2022). The platform exposes a simple API that allows developers to interface with its music repository, in which tracks are organized through specific features (Table 2).

Table 2. Audio features for a music track provided by the Spotify API.

Feature	Description	Range
Acousticness (a)	"A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic."	[0, 1]
Danceability (d)	"Danceability describes how suitable a track is for dancing based on a combination of musical elements, including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is the least danceable, and 1.0 is the most danceable."	[0, 1]
Energy (e)	"Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy."	[0, 1]
Instrumentalness (in)	"Predicts whether a track contains no vocals. 'Ooh' and 'aah' sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly 'vocal.' The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0."	[0, 1]
Liveness (liv)	"Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides a strong likelihood that the track is live."	[0, 1]
Loudness (lou)	"The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing the relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude)."	[-60, 0]
Speechiness (s)	"Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g., talk show, audiobook, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks."	[0, 1]
Tempo (t)	"The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, the tempo is the speed or pace of a given piece and derives directly from the average beat duration."	[30, 240]
Valence (v)	"A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g., happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g., sad, depressed, angry)."	[0, 1]

The Spotify API allows to interact with the repository in different ways, and it organizes the possible calls into several groups (Available online: <https://developer.spotify.com/documentation/web-api/reference/>, accessed on 18 October 2022).

3.3. Reinforcement Learning Notes

In real-world scenarios, individuals learn to make decisions based on their experience and interaction with the external environment. Such a learning process is related to the so-called law of effect: responses that produce a satisfying effect in a particular situation become more likely to occur again in that situation, and responses that produce a discomforting effect become less likely to occur again in that situation.

3.3.1. The Learning Model

In RL, the agent interacts with the environment by choosing from time to time which actions to take in order to achieve a goal. Each agent's action changes the environment's state and affects future choices within the environment itself. In order to monitor the not predictable effects of the actions, the agent takes into consideration some crucial elements: policy, reward, and value function. The policy is a mapping of the environment states into agent actions and essentially indicates which action is preferable to perform in correspondence with a particular state. The reward indicates how desirable it is, in the "immediate" term, for the agent to be in a specific state. In this sense, it can be intended as the "short-term" goal of the agent. The entire process is divided into a succession of actions by the agent over time, each corresponding to a change in the environment's state and a reward to the agent based on the action taken. The agent's main goal is to "maximize" the reward over time. The value function is the "long-term" goal for the agent. Given a state, the corresponding value function predicts the rewards determined from it, i.e., the total amount of reward that the agent will accumulate starting from it.

3.3.2. Markov Decision Processes

Markov Decision Processes (MDPs) can be used to provide a mathematical representation of the model described above since they are generally used for describing “decision making” contexts in which the decision maker affects the result of the decisions. We focus on environments in which the number of actions and states is finite (“finite” MDPs). Formally, an MDP is a tuple (S, A, P, R) , where (i) S is the set of states, (ii) A is the set of actions that the agent can undertake, (iii) P is the probability that action in some state s will result in the state s' , i.e., $P_a(s, s') = P(s' = S_{t+1} | S_t = s, A_t = a)$, (iv) R is the expected reward as a result of action a which led the environment to go from state s to state s' . An MDP searches for the “best policy” for the agent. A policy function π maps a pair (s, a) , where $s \in S$ is a state and $a \in A(s)$ is an action, to the probability $\pi(a|s)$ of undertaking a when in s . π is used for estimating a state the expected reward $V_\pi(s)$ when starting in s :

$$V_\pi(s) = E\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s\right]$$

We say that $V_\pi(s)$ is the state-value function for π . However, for a similar function based on the agent’s actions, it is necessary to relate the choice of action s under the policy π . So, we define the expected reward starting from s and taking the action a following π :

$$q_\pi(s, a) = E\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s, A_t = a\right]$$

3.4. Go-Explore

RL has made impressive progress in recent years, showing impressive performance in games such as Go [47,48]. However, these results hide some of the more difficult challenges not addressed in trying to scale RL to really complex situations, such as “hardexploration” problems: (i) sparse-reward problems, i.e., a large number of actions is necessary to obtain rewards; (ii) deceptive-reward problems, i.e., the reward feature provides misleading feedback, which can lead to getting stuck in a local optima.

4. Music Recommendation Based on Go-Explore

In this section, we propose Moodify, a music recommendation system based on GoExplore and able to induce target emotions in the user. Table 3 summarizes the main project decisions made to exploit the Go-explore paradigm for defining the proposed system.

Table 3. From Go-Explore to Moodify: project decisions.

Go-Explore	Moodify
The states correspond to the game’s cells (e.g., pixels)	The states correspond to the emotions
The agent begins by exploring the environment without any prior knowledge about it	The agent learns during the <i>training</i> phase based on user feedback without having to make any decisions
When the agent reaches a new state rewarding him with points, the algorithm stores such a state	When the agent reaches a new state rewarding him with points, the algorithm memorizes the musical features corresponding to that state
The agent continues to explore from a stored state, thus, being able to progress to new states over time	The agent continues to listen to memorized music, thus, being able to progress towards new emotions over time
Each time the game character dies, a negative reward is assigned to that cell	Whenever the user gives feedback that does not correspond to the desired emotion, the agent receives a negative reward

4.1. Preliminaries and Definitions

During a preliminary analysis, we observed that given an initial emotional state, the induction through music recommendation of a target emotional state practically never occurs through listening to a single piece of music. In fact, it is usually necessary to listen

to different songs with the passage of intermediate emotions. Formally, let E_s be the start emotional state and E_t be the target emotional state of a user, and let m_1, \dots, m_N be the sequence of musical songs that induces E_t in the user starting from E_s . We observed that $N \gg 1$. This interesting observation justifies using an RL-based approach to define an emotion-based music recommendation system able to induce emotions.

We define m_1, \dots, m_N as a musical trajectory from E_s to E_t for the user, and E_1, \dots, E_N the corresponding emotional trajectory from E_s to E_t for the user, where E_k is the intermediate emotional state induced in the user, after listening to m_1, \dots, m_k .

Observe that, let E be an emotional state, it corresponds to a specific point in the two dimensional space of the circumplex model shown in Figure 2. So, given two emotional states E_s and E_t , we say that the distance between E_s and E_t , denoted with $d(E_s, E_t)$, is the euclidean distance between the point in the circumplex model corresponding to E_s and the point in the circumplex model corresponding to E_t .

4.2. Problem Description

We emphasize that let E_s be the start emotional state and E_t be the target emotional state of a user, Moodify will be trained to propose the “best trajectory” from E_s to E_t . In our context, the concept of “best trajectory” is related to two aspects. On the one hand, we are interested in finding the musical trajectory that allows the listener to reach an emotional state that is “as close as possible” to the chosen target emotional state (appropriateness of the recommendation). On the other hand, we are interested in reaching the emotional state in the shortest possible time, therefore, in waiting as short as possible (responsiveness of the recommendation).

Thus, the problem faced at each request for a music recommendation can be formalized as follows: given a start emotional state E_s and a target emotional state E_t , the goal is to find the musical trajectory m_1, \dots, m_N , which, starting from E_s , (i) minimize the distance between E_t and E_t' , where E_t'

t is the target emotional state reached after listening m_1, \dots, m_N , and (ii) minimize the length N of the musical trajectory m_1, \dots, m_N . In the following, we

propose a Go-Explore-based system to face such a problem. The idea is to recommend music according to the best policy built by such a system. As we will see in Section 6, a preliminary evaluation study has been conducted to evaluate this approach.

4.3. The Methodology

This section provides details about the methodology followed to build Moodify. First, we will reformulate our decision-making context in terms of MDP. Then, we will describe the main steps of the proposed Go-Explore-based approach.

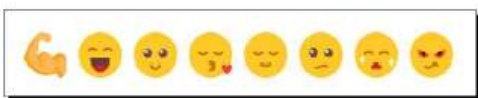


Figure 3. The emotional states are selectable in MoodifyWeb.

Formally, the complexity of the described MDP is:

$$\mathcal{O}(N_E \times \prod_x \text{range}(x) \times N)$$

4.3.2. Step 1: Listen until Solved

The goal is to discover high-performing trajectories in the emotion space, to be improved in Step 2. The result is an archive of different emotional states, named “cells”, and trajectories to follow to reach these states. This step is organized into several listening sessions. The goal of each session is to

find high-performing trajectories for one specific target emotional state chosen by the user. Indeed, at the beginning of the session, the user declares the state he would like to reach at the end of it, i.e., the target emotion he would like to experience after the listening session. At the beginning of each session, the archive only contains the initial emotion selected by the user (start state). From there, the system repeats the following steps: (i) select a cell from the current archive, (ii) explore from that cell location stochastically, i.e., recommend random

music and collect feedback from the user after the listening, (iii) add new cells and their trajectory to the archive. Here, we provide details about the cell, state representation, and the reward function based on the feedback the user provided at the end of each listening.

4.3.3. Step 2: Emotion Robustification

As a result of Step 1, Moodify collected a set of high-performing trajectories. To make the trajectories robust to any noise, Step 2 creates a policy via imitation learning. The idea is to build a policy that performs as well as the trajectory discovered during the exploration, but at the same time, it must be able to deal with circumstances not present in the original trajectory. As proposed in [19], to train the policy, we chose a Learning from Demonstration algorithm that proved to be able to improve upon its demonstrations, i.e., the Backward Algorithm. It works as follows: (i) the agent starts near the trajectory's last state t and

runs a standard RL algorithm (in our approach, we chose the Q-Learning approach) from such a state, (ii) when the algorithm has learned to get a better reward than t , the algorithm repeats the process by starting from a point near to the trajectory and repeats the process, (iii) if for each trajectory from the initial state the agent is able to obtain a better score, then stop the process.

4.3.4. Limitations of the method

At the end of the training, if E_1, \dots, E_N are the emotions on which the system has been trained, then Moodify has N Q-Table available, each of which will be used when the corresponding emotion is used as target emotion. The problem with this approach is that when we face complex environments, such as the emotional space described by the circumplex model shown in Figure 2, where the number of states and actions can grow, then Q-tables can become unfeasible. As we will see in Section 5, in this preliminary work, we focused on only 8 emotional states. However, as also highlighted in Section 7, in future work, we have planned to exploit Deep Q-learning techniques. Such techniques exploit power deep feedforward neural networks for computing Q-value, i.e., to use the output of such neural networks to get new Q-value.

5. MoodifyWeb: The Web Application

We developed a Web application, namely MoodifyWeb, which uses the method described in Section 4 to enable listening to music songs from Spotify according to target emotions selected by the user. For the development, we used Vue.js (Available online: <https://vuejs.org/>, accessed on 18 October 2022), a JavaScript framework and the Spotify API for developers (Available online: <https://developer.spotify.com/>, accessed on 18 October 2022). More details on the architecture and technologies involved are available in Appendix A.



Figure 4. MoodifyWeb user interface with main functionalities. (a) The login page. Here the user accesses our application. The access can be made with Facebook, Spotify, or Google accounts; (b) The emotion selection page. Here the user selects his/her current emotion and the target emotion; (c) The music recommendation page. Here the song to listen to is recommended, and the user can give feedback to the system in terms of which emotions have been felt; (d) The profile page. The user can update personal information.

6. Moodify's Assessment by Users

In this section, we show the preliminary evaluation study performed to evaluate Moodify in terms of effectiveness and overall user satisfaction, with a methodology already applied in earlier works [20]. We have

recruited 40 participants split among musicians (35%) and not musicians (65%). Participants used the machines available in our Lab, i.e., desktop computers equipped with Intel i7 quad-core processor and 16 GB RAM DDR3. The participants' sample was 60% male and 40% female, with a mean age of 24 (Standard Deviation = 4.76). Participants were informed that the information provided remains confidential. In Table 4, we report further details on the participants.

Table 4. Participant's demographics

Participants	Number	Percentage
Gender		
Male	24	60.0%
Female	16	40.0%
Age		
15-20 years old	8	20.0%
20-30 years old	28	70.0%
30+ years old	4	10.0%
Time spent on listening music per week		
<1 h	4	10.0%
1-3 h	12	30.0%
3+ h	24	60.0%

6.1. Method

The study included several steps: a Preliminary Step, a Testing Step, and a Summary Step, as defined in . In the Preliminary Step, participants had to fill in a question-based assignment concerning demographic and background information. The Testing Step's goal is to assess the experiences recommended during an experience time window, including ten listening sessions. It lasted roughly twenty days, with the longest session, which took two days. During the sessions, participants dealt with MoodifyWeb, thus, giving feedback concerning 3 usability aspects: (i) "Appropriateness of the music", which measures the perceived effectiveness of the system's behavior (recommended music), (ii) "User satisfaction", which measures the satisfaction of the users using your system, (iii) "System responsiveness", which measures the responsiveness of the system. Users expressed their feedback by answering question items on a 5-point Likert scale (from "Strongly disagree" to "Strongly agree").

6.2. Results

The Preliminary Step's results led us to outline a profile of the participants involved. Specifically, it sheds light on the users' habits. Indeed, they have the propensity to listen to music with the aim of experiencing a specific emotion, which in most cases is relaxing (70% answered more than once per day). Moreover, 35% of them have thought to use a specific tool designed for "recommending music" and "taking into account music preferences". Furthermore, users involved had a high familiarity with a variety of music platforms, such as Spotify and YouTube.

Table 5. Results of the Testing phase across all participants (average scores)

Recommendation	Appropriateness of the Recommendations	User Satisfaction	System Responsiveness
#1	4.15	4.25	4.45
#2	4.20	4.15	4.05
#3	3.85	4.15	3.90
#4	4.20	4.00	3.95
#5	4.25	4.15	4.10
#6	4.15	4.05	4.25
#7	3.95	3.90	4.15
#8	4.25	4.15	4.05
#9	4.10	4.15	3.90
#10	4.25	4.30	4.15

7. Conclusions

In the digital age, emotion/mood has become a fundamental criterion used by ICT systems in predicting social behaviors or conditioning people in their social interactions and work activities. In light of this, music systems that regulate mood and emotions in our daily life are arousing particular interest. Therefore, the affective computing research community has put efforts into modeling the relationship between music and emotion. Applications of affective computing studies can be found in education, health care, entertainment, affective ambient intelligence, multimedia retrieval, and music retrieval and generation. As for the specific musical context, most of these works consist of context-sensitive

recommendation tools which take into account the emotional state of the listener. Few results have been obtained in the study of music systems for induction of emotional states, i.e., methods to influence the emotional state of listeners and adapt interaction with technology to their affective state.

References

1. Grand View Research. Music Streaming Market Size, Share & Trends Analysis Report By Service (On-demand Streaming, Live Streaming), By Platform (Apps, Browsers), By Content Type, By End-use, By Region, And Segment Forecasts, 2022–2030, 2022. Available online: <https://www.grandviewresearch.com/industry-analysis/music-streaming-market> (accessed on 18 October 2022).
2. Hanjalic, A.; Xu, L.Q. Affective video content representation and modeling. *IEEE Trans. Multimed.* **2005**, *7*, 143–154. [[CrossRef](#)]
3. Lu, L.; Liu, D.; Zhang, H.J. Automatic mood detection and tracking of music audio signals. *IEEE Trans. Audio Speech Lang. Process.* **2005**, *14*, 5–18. [[CrossRef](#)]
4. Yang, Y.H.; Chen, H.H. Ranking-based emotion recognition for music organization and retrieval. *IEEE Trans. Audio Speech Lang. Process.* **2010**, *19*, 762–774. [[CrossRef](#)]
5. Yang, Y.H.; Chen, H.H. Machine recognition of music emotion: A review. *ACM Trans. Intell. Syst. Technol. (TIST)* **2012**, *3*, 1–30. [[CrossRef](#)]
6. Lara, C.A.; Mitre-Hernandez, H.; Flores, J.; Perez, H. Induction of emotional states in educational video games through a fuzzy control system. *IEEE Trans. Affect. Comput.* **2018**, *12*, 66–77. [[CrossRef](#)]
7. Muszynski, M.; Tian, L.; Lai, C.; Moore, J.; Kostoulas, T.; Lombardo, P.; Pun, T.; Chanel, G. Recognizing induced emotions of movie audiences from multimodal information. *IEEE Trans. Affect. Comput.* **2019**, *12*, 36–52. [[CrossRef](#)]
8. Juslin, P.N.; Sloboda, J.A. *Music and Emotion: Theory and Research*; Oxford University Press: Oxford, UK, 2001.
9. Zentner, M.; Grandjean, D.; Scherer, K.R. Emotions evoked by the sound of music: Characterization, classification, and measurement. *Emotion* **2008**, *8*, 494. [[CrossRef](#)]
10. Gabrielsson, A. Emotion perceived and emotion felt: Same or different? *Music. Sci.* **2001**, *5*, 123–147. [[CrossRef](#)]
11. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [[CrossRef](#)]
12. Paul, D.; Kundu, S. A survey of music recommendation systems with a proposed music recommendation system. In *Emerging Technology in Modelling and Graphics*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 279–285.
13. Agrafioti, F.; Hatzinakos, D.; Anderson, A.K. ECG pattern analysis for emotion detection. *IEEE Trans. Affect. Comput.* **2011**, *3*, 102–115. [[CrossRef](#)]
14. Shantala S Yalawar K Vijaya Babu, Mrutyunjaya S Yalawar, A New Approach for Secure Login Method and Forestall Cyber bullying in Social Media, 2019/5, *International Journal of Recent Technology and Engineering (IJRTE)*, Volume 8, Issue Ic2,1243-1246. https://scholar.google.co.kr/citations?view_op=view_citation&hl=en&user=Tyqm5YUAAA&citation_for_view=Tyqm5YUAAA:UebtZRa9Y70C
15. L. Pullagura, N. Kittad, G. Diwakar, V. Sathiya, A. Kumar and M. S. Yalawar, "ML based Parkinson's Disease Identification using Gait Parameters," 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2022, pp. 561-566, doi: 10.1109/ICACRS55517.2022.10029281.
16. Kumar, S., Sai Lavanya, G.V.R. (2020). Face Recognition Using Open CV with Deep Learning. In: Kumar, A., Paprzycki, M., Gunjan, V. (eds) *ICDSMLA 2019. Lecture Notes in Electrical Engineering*, vol 601. Springer, Singapore. https://doi.org/10.1007/978-981-15-1420-3_122

17. Mrutyunjaya S Yalawar, Basava S Dhanne, Rakesh Ranjan, Telugu Satyanarayana, Design and development of E-care management system for hospitals, 2019, Computing, Communication and Signal Processing: Proceedings of ICCASP 2018, 547-555, Springer Singapore.
18. Dr. Sheo Kumar, **Dr. Rajesh Tiwari** and NeethuChoudhary, “Detection of Phishing Attacks in Web Environment using Unsupervised Machine Learning” National Conference on Computational Methods, Data Science and Applications (NC-CMDSA 2021) held at Maulana Azad National Urdu University, Hyderabad on 24th – 25th May 2021.
19. Mahendra Kumar Sinha, **Rajesh Tiwari**, Manisha Sharma and Kamal K. Mehta, “Feature Extraction Techniques from Digital Image of OMR using Histogram with SIMD”, International Journal of Advance Research in Computer Science and Software Engineering, Vol. 5, Issue 6, June 2015, pp 1373 – 1381, ISSN: 2277 – 128X.
20. Suriya Begum, Farooq Ahmed Siddique, **Rajesh Tiwari**, “A Study for Predicting Heart Disease using Machine Learning”, Turkish Journal of Computer and Mathematics Education, Vol. 12, Issue 10, 2021, pp 4584-4592, e-ISSN: 1309-4653