DJ-Running: An Emotion-based System for Recommending *Spotify* Songs to Runners

P. Álvarez[1][a], A. Guiu[1], J. R. Beltrán[2][b], J. García de Quirós[1] and S. Baldassarri[1][c]

[1]*Department of Computer Science and Systems Engineering, University of Zaragoza, Zaragoza, Spain*

[2]*Department Electronic Engineering and Communications, University of Zaragoza, Zaragoza, Spain*

Keywords: Running, Music Recommendations, Runners' Emotions, Motivation and Performance.

Abstract: People that practice running use to listen to music during their training sessions. Music can have a positive influence on runners' motivation and performance, but it requires selecting the most suitable song at each moment. Most of the music recommendation systems combine users' preferences and context-aware factors to predict the next song. In this paper, we include runners' emotions as part of these decisions. This fact has forced us to emotionally annotate the songs available in the system, to monitor runners' emotional state and to interpret these data in the recommendation algorithms. A new next-song recommendation system and a mobile application able to play the recommended music from the *Spotify* streaming service have been developed. The solution combines artificial intelligence techniques with Web service ecosystems, providing an innovative emotion-based approach.

1 INTRODUCTION

Most of the people that practice running as a sport, listen to music during the development of the physical activity using their portable music players, mainly, mobile phones. Running with music can help to increase the runner's motivation, making hard training sessions much more pleasant as well as make the runner feel less alone. These effects are of special interest for long-distance runners, or even people with a sedentary lifestyle who wish to start running. Since music has the ability to produce reactions in the individuals (Terry, 2006), it cannot be ruled out that the listening of different types of music could also influence the motivation and performance of a runner (Brooks and Brooks, 2010). Therefore, a runner should select ideally a different playlist for each session depending on the type of training, his/her musical preferences, his/her mood, or the characteristics of the route he/she wants to run.

Music Recommendation Systems (MRS) reduce the effort of users for selecting songs by considering their profiles and preferences. Traditionally, these systems have been used to create personalised playlist that were then played on portables music players. The rise of music streaming services has motivated a new generation of recommendation systems more dynamic and flexible than traditional approaches, for example, *next-song recommendation* systems (Zheng et al., 2019; Baker et al., 2019; Vall et al., 2019b). These MRS create playlists in real time, while the user is listening them, and learn from users' short-term and long-term behaviors for improving their predictions. Additionally, the widespread use of mobile phones as players allows these systems to collect data about the user's context and utilize this contextual information to better satisfy users' interests (Kaminskas and Ricci, 2011; Wang et al., 2012; Chen et al., 2019). Despite the advances in developing context-aware systems, certain contextual decision factors are still underused, such as users' emotions and the activity that they are carrying out at each moment, for instance.

DJ-Running is a research project that monitors the runners' emotional and physiological activity during the training sessions, to automatically recognize their feelings and to select, in real-time, the most suitable music to improve their motivation and performance (DJR, 2019). In this paper, we present the context-aware music recommendation system developed as part of the project. It predicts the next song to be played considering the user's location and emotions

[a] https://orcid.org/0000-0002-6584-7259
[b] https://orcid.org/0000-0002-7500-4650
[c] https://orcid.org/0000-0002-9315-6391

Álvarez, P., Guiu, A., Beltrán, J., García de Quirós, J. and Baldassarri, S.
DJ-Running: An Emotion-based System for Recommending Spotify Songs to Runners.
DOI: 10.5220/0008164100550063
In *Proceedings of the 7th International Conference on Sport Sciences Research and Technology Support (icSPORTS 2019)*, pages 55-63
ISBN: 978-989-758-383-4

and the type of training session that he/she is carrying out. This combination of contextual factors is a contribution with respect to existing next-song recommendation proposals. In addition to the system, we also present the mobile application that monitors the runner's activity and connects with the *Spotify* music streaming service to play the recommended songs. From a technological point of view, the solution has been programmed combining the paradigms of cloud, fog and service computing.

The remainder of this paper is organized as follows. In Section 2, the related literature regarding to next-song recommendation systems is reviewed, paying attention in those solutions that are integrated with the *Spotify* services. Sections 3 and 4 describe the functionality and architecture of the *DJ-Running* application and recommendation system, respectively. The system consists of a set of Web services that are detailed in Sections 5 and 6. Finally, Section 7 presents the paper's conclusions and makes suggestions for future research.

2 RELATED WORKS

2.1 *Next-song* Recommendation Systems

The task of this class of recommendation systems is to predict the next song that a user would like to listen to. These predictions are mainly based on different factors related to users and songs (Knees et al., 2019). From the users' point of view, factors can be classified as intrinsic or external. The former are related to users' profile and preferences, listening histories and cultural features; whereas the external are mainly associated with the user's context, including the location (Kaminskas and Ricci, 2011; Su et al., 2010), the social context (Chen et al., 2019), the activity (Wang et al., 2012; Oliver and Kreger-Stickles, 2006) or the mood (Han et al., 2010), for instance. These contextual factors are more dynamics and have recently gained relevance thanks to users listen to music on their smart mobile phones. Today, these devices integrate rich sensing capabilities that allow to collect a wide variety of data about users' context and, as a result, most of the current recommendation systems make context-aware predictions (Wang et al., 2012). On the other hand, songs' features have been also used by these systems (Cano et al., 2005; Borges and Queiroz, 2017; Vall et al., 2019a; Zheng et al., 2019), specially songs' audio features. The interest in these approaches lies in their recommendations are based on data that can be extracted from songs and, therefore, are directly available. This availability alleviates the *cold start* problem, which arises when there is not data about a new user and, therefore, it is not possible to make effective recommendations (Chou et al., 2016).

Runners' emotions and types of training sessions are two factors considered as part of the *DJ-Running* recommendation system. In the field of affective computing many works propose methods to determine the emotions perceived by the user while listens to a song (Yang et al., 2018). These methods enhance songs' metadata by adding them emotional labels. Nevertheless, next-song recommendations systems do not use these labels to improve their results according to users' mood, for instance. Alternatively, the activity that the user is performing at that instant is another relevant factor, as it was discussed in (North et al., 2004; Levitin et al., 2007). Nevertheless, it is usually ignored by recommendation systems. As an exception, (Wang et al., 2012) proposes a solution that infers automatically a user's activity (working, sleeping, studying, running, etc.) and recommends songs suitable for that activity, and (Oliver and Kreger-Stickles, 2006) creates playlists to help users to achieve their exercise goals (walking or jogging, for example).

The techniques used to recommend the next song are varied, even most of them are those commonly used in general purpose recommendation systems: the collaborative filtering (Lee et al., 2010; Vall et al., 2019a; Baker et al., 2019; Chen et al., 2019), the content-based filtering methods (Pampalk et al., 2005; Cano et al., 2005; Oliver and Kreger-Stickles, 2006) and the hybrid approaches that combine both techniques to mitigate their possible drawbacks (Vall et al., 2019b). The collaborative techniques focus on the analysis of the similarity and the relationships between users. It is becoming more frequent that the friendship between social network users is included in the models of these collaborative approaches (Chen et al., 2019). Alternatively, the content-based methods are mainly based on songs' features and users' preferences and contexts. Nevertheless, nowadays the tendency is to integrate different techniques to take advantage of the strengths of each of them. These hybrid proposals also combine the common recommendation techniques with probabilistic models (Wang et al., 2012; Borges and Queiroz, 2017) or different classes of neuronal networks (Choudhary and Agarwal, 2017; Zheng et al., 2019), for instance.

On the other hand, there is a growing interest in solving the problem of automating the *music playlist continuation* (Oliver and Kreger-Stickles, 2006; Vall et al., 2019a; Baker et al., 2019). It consists of providing a personalised extension to the playlist that the user is listening to. Therefore, it is a particular case of next-song approaches in which the recommendations are made by considering the songs previously played (each playlist is processed as a user's listening history). The factors that are usually involved in these recommendations are songs' order and popularity (Vall et al., 2019b), the songs' audio features (Vall et al., 2019a; Baker et al., 2019), the most listened authors and musical genres, or users' response to recommended songs (Pampalk et al., 2005; Oliver and Kreger-Stickles, 2006; Choudhary and Agarwal, 2017), among others. This last factor gains relevance in these recommendation systems as an immediate and explicit feedback that helps to maximize the user's satisfaction. This feedback can be determined by using sensors to detect physical and physiological responses (Oliver and Kreger-Stickles, 2006), by evaluating the effects of playing the same list of songs in a different order (Choudhary and Agarwal, 2017), or by analyzing the user's dislikes (for example, the songs that are skipped by the user (Pampalk et al., 2005)).

2.2 Music Recommendation Systems based on *Spotify*

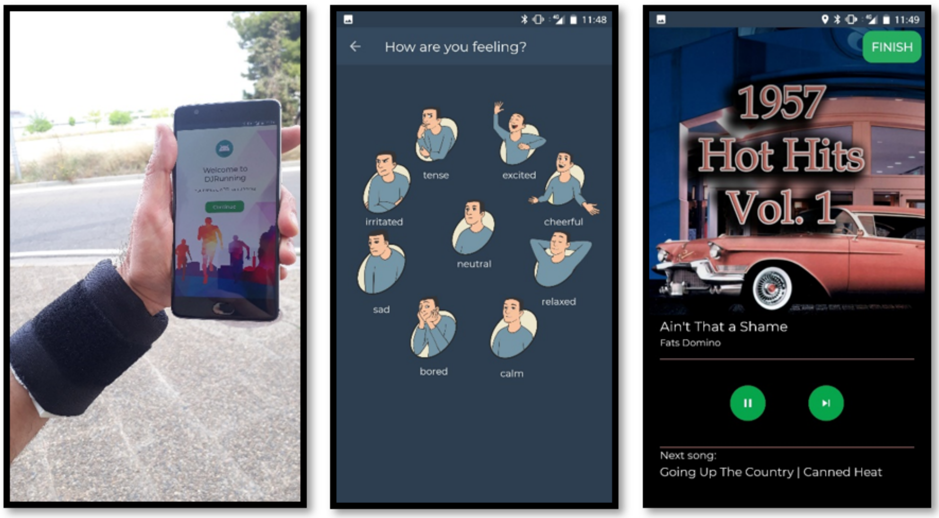
Most of the works related to *Spotify* propose music recommendation systems to help users to create their playlists. Recommendations are based on the user's preferences (musical genres and artists, mainly) and the features of songs that he/she usually listens to. Users' profiles are determined by utilizing users' past interactions (Fessahaye et al., 2019; Bennet, 2018) or by processing the messages published by users in social networks, such as *Twitter* (Pichl et al., 2015) or *Facebook* (Germain and Chakareski, 2013). Internally, these recommendation systems are programmed integrating content and collaborative filtering techniques (Fessahaye et al., 2019; Pichl et al., 2015; Germain and Chakareski, 2013). The first ones help to determine the similarity between songs based on their audio features, while the latter determine the similarity between users based on their preferences. The same approach is currently used by *Spotify* (Madathil, 2017). Exceptionally, (Bennet, 2018) makes recommendations using clustering techniques. As a conclusion, with the exception of (Fessahaye et al., 2019), these *Spotify*-based systems do not consider music emotions. (Fessahaye et al., 2019) adds a mood value to the set of songs' features which is obtained from the *Million Playlist Dataset*, released by *Spotify* in 2018 as part of the *Spotify RecSys Challenge*.

3 THE DJ-Running APPLICATION

The *DJ-Running* technological infrastructure allows a runner to configure his/her profile in order to listen personalized music during the training sessions. The runner accesses this functionality by a mobile application that works with the *Spotify* streaming service to play the recommended songs. The aim of this section is to describe the design of this application and its interaction with the user and the environment.

Before using the mobile application for the first time, the runner sets up his/her personal profile through a Web application. This profile includes anthropometric characteristics, demographic data, musical preferences, the type of runner, etc. It is highly recommended to update this information periodically. On the other hand, the user also has to purchase a license of *Spotify*. When the application is installed in the mobile phone the first time, it requires the user's credentials created by the Web application (to access to the information of the profile) and the license code of *Spotify*. Once the software installation is completed, the application is ready to be used during the training sessions. Figure 1-a shows the initial application screen. Before starting to run, the runner must introduce the kind of training session that he/she will do and his/her emotional state at that moment (happy, relax, stressed, etc.). The screen in which the runner is asked about the emotional state is presented in Figure 1-b, and it is based in the *'Pick-A-Mood' (PAM)* model (Desmet et al., 2012), a cartoon-based pictorial instrument for reporting and expressing moods. This model classifies the possible emotional states of the user according to the reference model of affect proposed by *Russell* (Russell, 1980). Optionally, the application can also be connected to an emotional wearable developed in the frame of our project (Álvarez et al., 2019). This device incorporates a set of sensors (GSR, HR and oximeter) that allow to detect the runner's emotional state during the training session. These estimations are carried out by an artificial intelligent system that translates the low level signals of the sensors into emotions. After this initial configuration stage, the application starts to play personalized music by the streaming of *Spotify*. Figure 1-c shows the interface of the application while it is playing the music. The runner can stop the player, skip the song (this actions is considered for future recommendations) o stop the training session (top right of the screen). During this stage, the application also monitors the runner's geographic location.

The decision about which song must be played at each moment is complex. The *DJ-Running* recommendation service is in charge of taking this deci-



(a) (b) (c)

Figure 1: Interface of the *DJ-Running* mobile application.

sion based in three different types of input parameters: the runner's profile (set when he/she registered in the system), the data explicitly introduced by the runner before starting the training session (kind of training and emotional state), and, lastly, the data automatically recorded during the sport activity (runner location, his/her heart rate, running pace and the changes experienced in his/her emotional state, mainly). Furthermore, the recommendation system also interacts with the *Spotify* services to access to the musical preferences of the user, and with different geographic systems that offer relevant information related with the runner's current location (meteorological information, level of noise, kind of terrain, altimetry, etc.). The system takes into account all this information to determine, in a personalized way, the next song to play. The technological components that are involved in this complex process are described in detail in the following section.

Finally, when the runner ends the training session, the application shows in a *Google Maps* map the route followed and a summary of the session activity.

4 ARCHITECTURE AND DEPLOYMENT OF THE *DJ-Running* SYSTEM

Figure 2 shows the software architecture of the *DJ-Running* system. Once the runner has signed up into the system, he/she connects with the music recommendation service using the *DJ-Running* mobile application (right part of the figure). Internally, the architecture consists of a set of Web services that provide the functionality needed to create users' profiles, to access music information and geographic data, and to make personalised music recommendations. These services require to interact with other external providers to offer the mentioned functionality, for example, with *Spotify*, *Acoustic-Brainz*, *Weather Underground* or some *Spatial Data Infrastructures* (left part of the figure).

The *Music recommendation service* is the core of the *DJ-Running* system. It receives the recommendation request from the runner's mobile application and predicts the next song to play. These recommendations are made considering the runner's context information, the runner's profile and emotionally-labelled songs available in the system. Most of these data are accessible through the *DJ-Running* data services: the *User Profile service*, the *Geo Data service* and the *Music Delivery service*. On the other hand, the *Music data service* integrates a *Music Emotion Recognition* systems (MER) that has been used to annotate emotionally *Spotify* songs. A complete description of all these services is presented in the next section.

From a technological point of view, the *Spring framework* has been used for developing all these services (Pivotal Software, 2017). It provides the core features needed for programming, configuring and deploying any Java-based service. More specifically, these services have been implemented as RESTful Web services in order to facilitate their integration in Web-accessible applications. On the other hand, the *Kubernetes* technology has been also used to au-

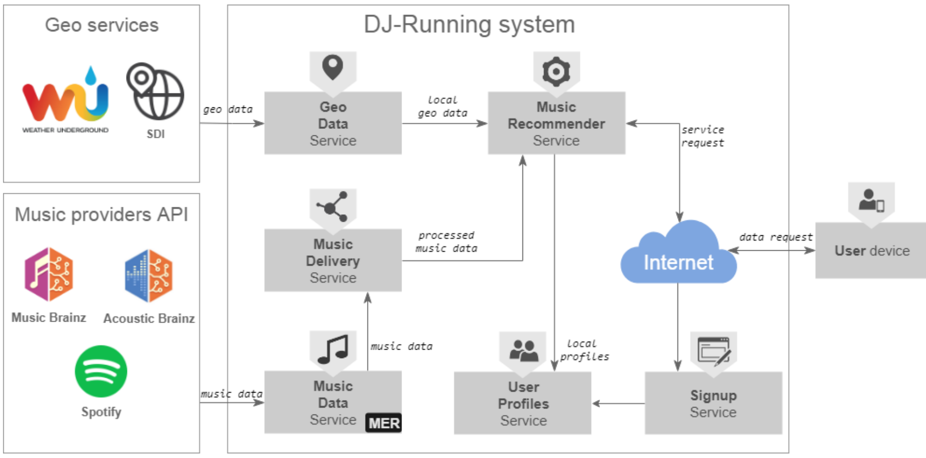


Figure 2: Architecture of the *DJ-Running* system.

tomate and manage the deployment of services in cloud-based execution environments. Ideally, these deployments should consider runners' location for reducing the communication latency and improving the quality of service, in particular the deployment of the music recommendation service. The fog computing paradigm (Mahmud et al., 2018) has been applied to fulfill these deploy requirements.

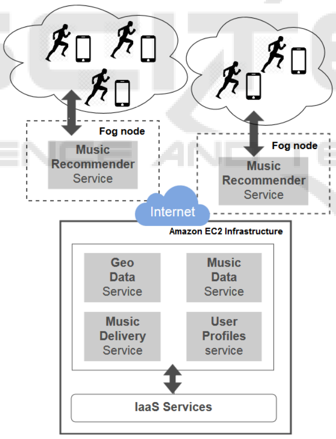


Figure 3: Deployment of the *DJ-Running* services.

Figure 3 shows the deployment of *DJ-Running* system. Data Web services have been deployed and executed in a public cloud provider, more specifically, in the *Amazon EC2 infrastructure*. These services are the back-end of the system and do not interact directly with runners and their applications. The music recommendation service is the front-end of the system and, therefore, it must reside at the edge of the network (or fog node). We have programmed the recommendation system as a lightweight service that can be deployed into a node with low computing and storage capacity, for example, into a *Raspberry Pi 3* computer. In our deployment, this small-size computer has been connected to a network close to the area where the runners who participate in the system test-beds ran. Additionally, different instances of the recommendation service could be easily configured and deployed into geographically distributed servers to cover a wider geographical area.

5 WEB DATA SERVICES

In this section the functionality of the Web data services and the most relevant implementation decisions are detailed.

5.1 The User Profile Service

An user model has been defined to create a digital representation of runners. It includes runners' anthropometric characteristics (height, weight, etc.), demographics data (age, genre, education, etc.), musical preferences (musical genres, authors, popularity, etc.) and social relationships. Most of these data are currently completed by the runner as part of the sign-up process, and can be updated via the Web application at any moment. Others can be automatically gathered by the system. For example, musical preferences can be continuously updated from the user's *Spotify* listening histories, or friendships can be determined from the user's social network account (such as, *Twitter* or *Facebook*). Both gathering processes require the user' extra information and permissions. On the other hand, the model also includes the songs that are skipped by an runner as part of the musical preferences. This feedback is sent to the system by the runner's mobile application during training sessions.

The *User Profile service* stores runners' descriptions defined from the previous model and manages the gathering processes needed to update them. Its Web API provides access to these descriptions, offering a set of search operations. This functionality is mainly used by the music recommendation service.

5.2 Services for Accessing to Music Information

Two services are responsible for managing and providing access to the collection of songs available in the *DJ-Running* system. The *Music Data service* interacts with the *Spotify Web API for developers* in order to retrieval songs of interests by applying different criteria (songs' popularity, musical genres, or artists, for instance). Metadata and audio features of these songs are requested to the music provider and stored them locally into the service's database. Subsequently, these songs are annotated emotionally by a *Music Emotion Recognition* system (MER). It integrates a set of machine learning models able to determine the emotion perceived by an user when listening to a song from its audio features. These emotions are represented by means of labels that are stored jointly with song's metadata into the service's database. The current version of our annotated database contains over 60,000 popular songs. Nevertheless, the system has been developed for automatically processing a massive collection of songs by providing an alternative to other approaches based on evaluations with users or experts.

In this work, the emotional labels are based on the Russell's circumplex model (Russell, 1980), one of the most popular dimensional models in affective computing. It represents affective states over a two-dimensional space that is defined by the valence (X-axis) and arousal (Y-axis) dimensions. The valence represents the intrinsic pleasure/displeasure (positive/negative) of an event, object or situation, and the arousal the feeling's intensity. The combination of these two dimensions (valence/arousal) determines four different quadrants: the aggressive-angry (negative/positive), the happy (positive/positive), the sad (negative/negative) and the relaxed (positive/negative) quadrant. In our proposal the emotional annotation of a song can have one of these four values: *Angry*, *Happy*, *Sad* and *Relaxed*. This annotation represents the emotion perceived by the users and corresponds with one of the Russell model's quadrants.

On the other hand, the *Music Delivery service* provides the recommendation system with access to the songs stored in the database. Internally, it creates a set of indexes to improve the efficiency of searches in large-scale datasets of emotionally annotated songs. These indexes were built considering the decision rules that guide the recommendation process of songs. Additionally, the service provides a wide variety of search criteria and options to facilitate the access to songs of interest.

5.3 The Geographic Data Service

Some geographic data related to the runner's location are used as contextual factors by the *DJ-Running* recommendation service. We are mainly interested in meteorological information and data that describe the environment in which the user is running. The *Geo Data Service* is responsible for interacting with Web services that provide these data as well as updating periodically a local database in which the geodata of interest are stored. Some of these data are changeable and, therefore, they must be constantly updated (the meteorological, noise-level or pollution data, for instance); whereas, others have a more static nature (the type of terrain, the average slope of terrain, the altitude, etc.).

Currently, the service interacts with the *Weather Underground* API to get meteorological information at real time. Besides, it integrates the set of services published by the *Spanish Spatial Data Infrastructure* (SDI). These enable the download and analysis of a wide variety of geographic data produced in Spain.

6 MUSIC RECOMMENDATION SERVICE

Figure 4 shows the high-level architecture of the recommendation service. The core component is the *recommendation system*. Its implementation is based on a *Nearest Neighbor Search* algorithm (NNS). This class of algorithms solve the problem of finding the point in a given set that is closest (or most similar) to a given point. Formally, they are defined from a set of points in a *space M* and a *metric distance* that allows to determine the similarity (or dissimilarity) between these points.

In our proposal, each *Spotify* song has been translated to a point of the space M. A point is a numeric vector created from the song's audio features and emotional labels. Before starting the service execution, these points have been calculated from songs available into the *Music Delivery service*, and stored them into a local repository (it represents the *space M*). Different indexes have been also created to improve the performance of executing the metric distance on the repository of points.

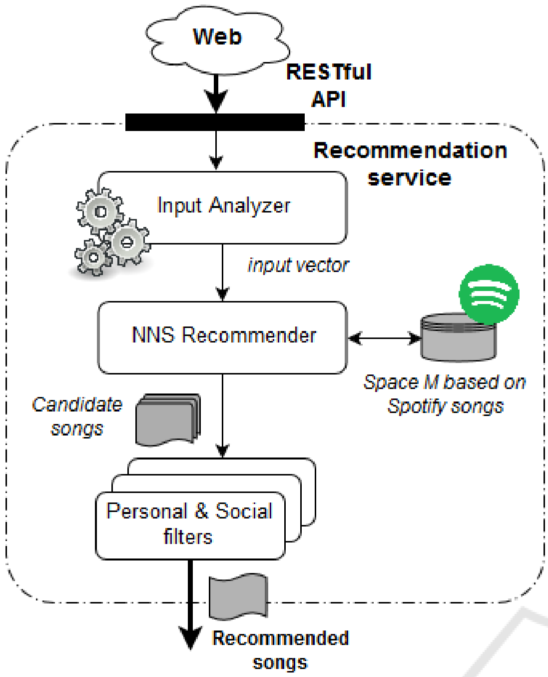


Figure 4: *DJ-Running* Music Recommendation system.

Despite these indexes, the search space of our problem is complex (*Spotify* stores more than 30 million of songs and a point is defined for each of these songs). Besides, it is not necessary to retrieve an exact search result, which is an overkill in this type of applications. Therefore, we have decided to use the *Annoy* algorithm (Erik Bernhardsson, 2013), an *approximate NNS* algorithm that has provided good results in the field of multimedia recommendations. These algorithms are able to retrieve approximate nearest neighbors much faster than NNS algorithms. Internally, our implementation of *Annoy* was configured for using the *angular distance* as similarity measurement between the points. Its performance and accuracy were evaluated by executing the *ANN-benchmarks environment*, a benchmarking framework for approximate nearest neighbor algorithms search (Aumüller et al., 2018), by obtaining satisfactory results from the point of view of our problem.

The other two components of the recommendation system are the *Input analyzer* and a set of *Filters*. The former is responsible for determining what kind of song (or songs) will be recommended to the runner considering the service request's input parameters (the runner location and mood, the kind of training, the emotion to be induced to the runner, or level of fatigue, among others). A rule engine translates these parameters to a vector that describes the audio and emotional features of the song (or songs) to find in the search space. The emotional labels of interest are mainly determined from the geographic and environmental data related to the runner location and the emotions that the system tries to induce him/her. Finally, the resulting vector corresponds with the point in the space that will be submitted as input to the recommendation system.

On the other hand, a list of candidate songs is returned as output of a recommendation request. Currently, 50 songs are recommended for each input request (this number is easily configured). *Filters* are responsible for scoring and ranking the returned songs in order to personalize the final recommendations. Two class of filters have been implemented: *Personal filters* and *Social filters*. The first ones score the candidate songs in accordance with the runner's musical preferences. These preferences were specified by the runner when he/she registered in the system. The social filters are based on the concept of similarity between runners. This similarity has different dimensions: the range of age groups, the physical/emotional response to a kind of training, the musical preferences or the friendship in social networks (such as *Twitter* or *Facebook*), for instance. *Clustering* algorithms and *Collaborative filtering* techniques have been combined to implement a prototype version of these social filters. Once all these filters have been executed, the three best ranked songs are returned as result of the recommendation service.

Finally, as an example, we explain the kind of songs that would be recommended for a specific training session. The runner's profile is a man, middle-aged and half-marathon runner. He is in the final stage training previous to a race and, therefore, he needs to maintain a good running pace. The training session consists of three parts: 15 minutes easy run, then 10 x (1 minute fast, 1 minute jog) and, finally, 15 minutes easy run (a total of 50 minutes). Besides, the route is mostly flat to favour a good running pace, and the day is sunny and with a pleasant temperature. In this example, if the runner is having a good performance during the training, the input analyzer will propose the following kinds of songs. For the first part of the training, there are selected *happy* songs, but *not specially motivating*. Then, *motivating* and *relaxing* songs will be alternated for each of the 10 repetitions (for the fast and jog pace, respectively). And, finally, *calm* and *relaxed* songs will be again selected to make easier the runner's recovery during the last part of the session. Nevertheless, if the runner's performance is not as expected, alternative songs should be proposed (for example, *motivating* songs that help him to improve an undesirable running pace during the third part of the training as an alternative to *calm* and *relaxed* ones). In any of the cases, these emotional labels are determined by the analyzer at real-time and, later, used by the recommendation algorithm to determine the concrete songs to be played at each moment. As conclusion, these songs have been selected by considering the runner's profile, location and emotional state.

7 CONCLUSIONS AND FUTURE WORK

In this paper a next-song recommendation system for runners has been proposed. The system makes personalised recommendations to increase runners' motivation and performance. Different context-aware factors are considered as part of the recommendation process, specially, the runner's emotions and the kind of training session. The inclusion of users' emotions in this process has forced us to emotionally annotate *Spotify* songs using machine-learning techniques and, then, to interpret these annotations in the recommendation algorithms. The current version of the system consists of a mobile application able to play music from the *Spotify* streaming service and a set of Web services that provide the functionality needed to make context-aware recommendations.

The *DJ-Running* system is being currently validated with real users (more specifically, with triathletes) in collaboration with the Sports Medicine Centre of the Government of Aragón (Spain). We are interested in studying the runners' emotional and physical responses when they train with certain kind of music. Triathletes must complete the session of 50 minutes described in the previous section on two occasions (during two consecutive weeks, once a week). The first time, the system is configured for recommending songs that produce a positive effect in the runner. We have used the emotional labels described in the example for selecting the songs to be played. The second time, the goal is to produce a negative effect in the runner and, therefore, we have selected *noisy* and *aggressive-angry* songs instead of the ones proposed previously. Although this validation is still in progress, preliminary results show that the different kinds of songs affect runners' performance and perceive exertion.

As future work, validation results will be used to improve the interpretation of context-aware factors as well as the recommendations. Besides, we would like to enhance the proposed user model, to automate the processes for gathering registered users' data, and to improve the efficiency of recommendation algorithms. Our final aim is to publish a version of our application on *Google Play*.

ACKNOWLEDGEMENTS

This work has been supported by the TIN2015-72241-EXP project, granted by the Spanish Ministerio de Economía y Competitividad, and the DisCo-T21-17R and Affective-Lab-T25-17D projects, granted by by the Aragonese Government and the European Union through the FEDER 2014-2020 "Construyendo Europa desde Aragón" action.

REFERENCES

(2019). The *DJ-Running* project. https://djrunning.es/.

Álvarez, P., Beltrán, J. R., and Baldassarri, S. (2019). Dj-running: Wearables and emotions for improving running performance. In Ahram, T., Karwowski, W., and Taiar, R., editors, *Human Systems Engineering and Design*, pages 847–853, Cham. Springer International Publishing.

Aumüller, M., Bernhardsson, E., and Faithfull, A. (2018). Ann-benchmarks: A benchmarking tool for approximate nearest neighbor algorithms.

Baker, J. C., Averill, Q. S., Coache, S. C., and McAteer, S. (2019). Song recommendation for automatic playlistcontinuation. Digital WPI, Retrieved from https://digitalcommons.wpi.edu/mqp-all/6744.

Bennet, J. (2018). A scalable recommender system for automatic playlist continuation. Data Science Dissertation. University of Skövde, School of Informatics, Sweden.

Borges, R. and Queiroz, M. (2017). A probabilistic model for recommending music based on acoustic features and social data. In *16th Brazilian Symposium on Computer Music*, pages 7–12.

Brooks, K. and Brooks, K. (2010). Enhancing sports performance through the use of music. *Journal of exercise physiology online*, 13(2):52–58.

Cano, P., Koppenberger, M., and Wack, N. (2005). Content-based music audio recommendation. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 211–212. ACM.

Chen, J., Ying, P., and Zou, M. (2019). Improving music recommendation by incorporating social influence. *Multimedia Tools and Applications*, 78(3):2667–2687.

Chou, S.-Y., Yang, Y.-H., Jang, J.-S. R., and Lin, Y.-C. (2016). Addressing cold start for next-song recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 115–118.

Choudhary, A. and Agarwal, M. (2017). Music recommendation using recurrent neural networks.

Desmet, P., Vastenburg, M., Bel, D., V., and Romero, N. (2012). Pick-a-mood; development and application of a pictorial mood-reporting instrument.

Erik Bernhardsson (2013). Annoy. https://github.com/spotify/annoy.

Fessahaye, F., Pérez, L., Zhan, T., Zhang, R., Fossier, C., Markarian, R., Chiu, C., Zhan, J., Gewali, L., and Oh, P. (2019). T-recsys: A novel music recommendation system using deep learning. In *2019 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–6.

Germain, A. and Chakareski, J. (2013). Spotify me: Facebook-assisted automatic playlist generation. In *2013 IEEE 15th International Workshop on Multimedia Signal Processing (MMSP)*, pages 25–28.

Han, B.-j., Rho, S., Jun, S., and Hwang, E. (2010). Music emotion classification and context-based music recommendation. *Multimedia Tools and Applications*, 47(3):433–460.

Kaminskas, M. and Ricci, F. (2011). Location-adapted music recommendation using tags. In *Proceedings of the 19th International Conference on User Modeling, Adaption, and Personalization*, UMAP'11, pages 183–194, Berlin, Heidelberg. Springer-Verlag.

Knees, P., Schedl, M., Ferwerda, B., and Laplante, A. (2019). User awareness in music recommender systems. In *Personalized Human-Computer Interaction*.

Lee, S. K., Cho, Y. H., and Kim, S. H. (2010). Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations. *Information Sciences*, 180(11):2142–2155.

Levitin, D., D James Mcgill, P., Daniel, D., and Levitin, J. (2007). Life soundtracks: The uses of music in everyday life'.

Madathil, M. (2017). Music recommendation system spotify - collaborative filtering. Reports in Computer Music. Aachen University, Germany.

Mahmud, R., Kotagiri, R., and Buyya, R. (2018). *Fog Computing: A Taxonomy, Survey and Future Directions*, pages 103–130. Springer.

North, A. C., Hargreaves, D. J., and Hargreaves, J. J. (2004). Uses of music in everyday life. *Music Perception: An Interdisciplinary Journal*, 22(1):41–77.

Oliver, N. and Kreger-Stickles, L. (2006). Papa: Physiology and purpose-aware automatic playlist generation. In *ISMIR*, volume 2006, page 7th.

Pampalk, E., Pohle, T., and Widmer, G. (2005). Dynamic playlist generation based on skipping behavior. In *ISMIR*, volume 5, pages 634–637.

Pichl, M., Zangerle, E., and Specht, G. (2015). Combining spotify and twitter data for generating a recent and public dataset for music recommendation. In *Proceedings of the 26th GI-Workshop Grundlagen von Datenbanken (GvDB 2014)*, Ritten, Italy.

Pivotal Software (2017). Spring framework. https://spring.io/projects/spring-framework.

Russell, J. (1980). A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161–1178.

Su, J., Yeh, H., Yu, P. S., and Tseng, V. S. (2010). Music recommendation using content and context information mining. *IEEE Intelligent Systems*, 25(1):16–26.

Terry, P. (2006). Psychophysical effects of music in sport and exercise: an update on theory, research and application. pages 415–419.

Vall, A., Dorfer, M., Eghbal-zadeh, H., Schedl, M., Burjorjee, K., and Widmer, G. (2019a). Feature-combination hybrid recommender systems for automated music playlist continuation. *User Modeling and User-Adapted Interaction*, 29:527–572.

Vall, A., Quadrana, M., Schedl, M., and Widmer, G. (2019b). Order, context and popularity bias in next-song recommendations. *International Journal of Multimedia Information Retrieval*, 8(2):101–113.

Wang, X., Rosenblum, D., and Wang, Y. (2012). Context-aware mobile music recommendation for daily activities. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 99–108. ACM.

Yang, X., Dong, Y., and Li, J. (2018). Review of data features-based music emotion recognition methods. *Multimedia Syst.*, 24(4):365–389.

Zheng, H.-T., Chen, J.-Y., Liang, N., Sangaiah, A. K., Jiang, Y., and Zhao, C.-Z. (2019). A deep temporal neural music recommendation model utilizing music and user metadata. *Applied Sciences*, 9(4).