Mobile Music Genius: Reggae at the Beach, Metal on a Friday Night?

Markus Schedl
Johannes Kepler University
Linz, Austria
markus.schedl@jku.at

Georg Breitschopf
Johannes Kepler University
Linz, Austria
georg.breitschopf@gmail.com

Bogdan Ionescu
University Politehnica of Bucharest
Bucharest, Romania
bionescu@alpha.imag.pub.ro

ABSTRACT

The amount of music consumed while on the move has been spiraling during the past couple of years, which requests for intelligent music recommendation techniques. In this demo paper, we introduce a context-aware mobile music player named "Mobile Music Genius" (MMG), which seamlessly adapts the music playlist on the fly, according to the user context. It makes use of a comprehensive set of features derived from sensor data, spatiotemporal information, and user interaction to learn which kind of music a listeners prefers in which context. We describe the automatic creation and adaptation of playlists and present results of a study that investigates the capabilities of the gathered user context features to predict the listener's music preference.

Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems

Keywords

Music Information Retrieval, User-centric Music Recommendation, Automatic Playlist Generation

1. MOTIVATION AND RELATED WORK

The need to take into account contextual aspects of the user when elaborating systems for music recommendation or automated playlist generation has frequently been acknowledged [18, 13, 14, 6, 2]. This is of particular importance in a mobile music consumption scenario, in which the user context frequently changes. Nevertheless, most approaches to music recommendation create playlists based on collaborative filtering [3], audio features [15], web-based features [17], or a combination of these [12].

Context-aware approaches differ considerably in how the user context is defined, gathered, and incorporated. Some approaches consider only temporal features [5] or weather conditions [10], while others model the user in a more comprehensive manner, including intrinsic and extrinsic factors. Cunningham et al., for instance, investigate how various factors relate to music taste (e.g., human movement, emotional status, and external factors such as temperature and lightning conditions) and subsequently propose a fuzzy logic model to create playlists [6]. Baltrunas et al. suggest a context-aware music recommender for listening while driving [1]. Even though the authors take into account eight different contextual factors (e.g., driving style, mood, road type, weather, traffic conditions), their application scenario is rather restricted and their system strongly relies on explicit human feedback. Literature targeting a mobile usage scenario commonly suggests systems that aim at matching music with the current pace of a walker or jogger [13, 2]. Such systems typically try to match the user's heartbeat with the music played [11]. However, they usually require additional hardware for context logging (e.g., heart rate sensors or pedometers) [8, 7, 6].

In this paper, we introduce the "Mobile Music Genius"[1] (MMG) player which makes use of a wide variety of contextual aspects to create and adapt playlists during playback on mobile devices (Section 2). We further report on first experiments to quantitatively investigate the capability of user context features to predict individual music taste (Section 3). Eventually, we draw conclusions and point to future research directions (Section 4).

2. MOBILE MUSIC GENIUS

The MMG player is an intelligent mobile music player for the `Android` platform. It aims to dynamically and seamlessly adapt the music playlist according to the music preference of the user in a given context.

An overview of the entire system can be found in Figure 1. The main input to MMG is the user's music collection based on which music context data (collaborative tags) is gathered from `Last.fm` to create static playlists. In addition, a wide variety of user context data (detailed below) is continuously monitored while the user interacts with the player or just enjoys the music. Both music context and user context data are stored in a database on the device; the user context is further uploaded to a context server at fixed time intervals. Both types of data are then combined to enable adaptive playlist generation. More details are provided in [4].

From (1) the contextual user data, (2) implicit user feedback (play, pause, stop, skip events), and (3) meta-data

[1] `http://www.cp.jku.at/projects/MMG`

ICMR '14, Apr 01-04 2014, Glasgow, United Kingdom

Category	Exemplary Attributes
Time	day of week, hour of day
Location	provider, latitude, longitude, accuracy, altitude
Weather	temperature, wind direction and speed, precipitation, humidity, visibility, pressure, cloud cover, weather condition
Device	battery level and status, available internal/external storage and memory, volume settings, audio output mode
Phone	service state, signal strength, GSM indicator, network type
Task	recently used tasks (apps), screen state (on/off), docking mode
Network	mobile network: available, connected; Wi-Fi: enabled, available, connected, BSSID, SSID, IP, link speed, RSSI; active network: type, subtype, roaming; Bluetooth: available, enabled
Ambient	light, proximity, temperature, pressure, noise
Motion	acceleration force, orientation of user, orientation of device, rate of rotation
Player	repeat mode, shuffle mode, APM mode, sound effects

Table 1: User context attributes monitored by MMG.

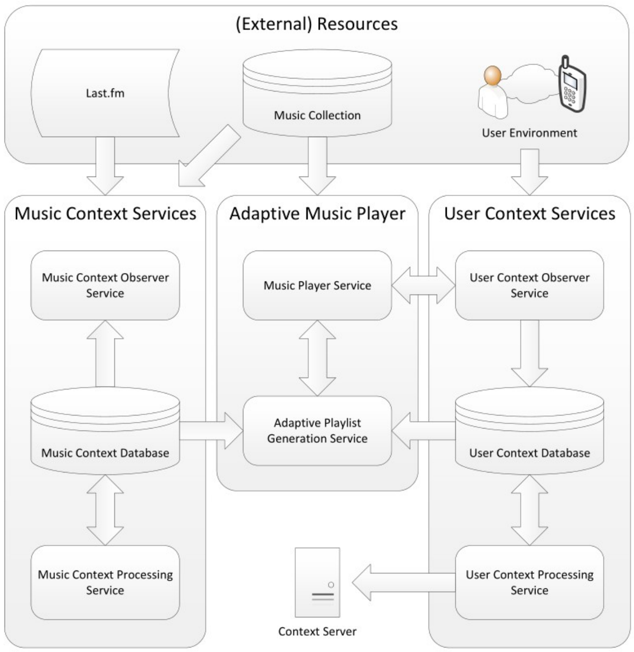

Figure 1: Overview of the MMG system.

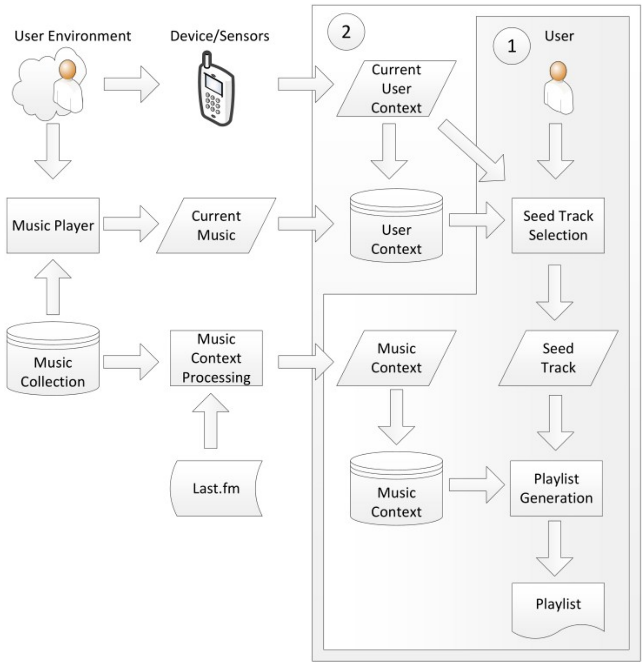

Figure 2: Overview of playlist generation in MMG.

about the music itself (artist, album, track, genre), MMG learns relationships between (1) and (3), i.e., which kind of music the user prefers in which situation. The underlying assumption is that music preference changes with user context. A user might, for instance, want to listen to an agitating rock song when doing outdoor sports, but might prefer some relaxing reggae music when being at the beach at a sunny and hot day. Table 1 lists the gathered context features. In addition to this unobtrusively acquired information, we allow but not require the user to enter her activity and mood each time a new track starts, presuming that both influence music taste, but are not easy to derive with high accuracy from the attributes listed in Table 1.

MMG supports three different types of playlist generation or adaptation: (1) seed-based via semantic tags, (2) user context-based seed selection, and (3) adaptive playlist modification based on user context. An overview of (1) and (2) is given in the right-hand part of Figure 2; (3) is depicted in the center of Figure 1.

2.1 Playlist Creation via Semantic Tags

To create playlists based on a seed song which is selected by the user, we compute similarities on term weights inferred from collaborative tags. A user-adjustable number of tracks closest to the seed is then added to the playlist. To this end, MMG first gathers the top tags for each song and each artist in the user's music collection, using the `Last.fm` API[2]. These top tags are then indexed using a predefined dictionary of music genres and moods in order to account for the limited computing capabilities of mobile devices and to keep constant the dimensionality of the feature vectors. As a result, each artist and track is represented by a 3,290-dimensional term weight vector w_a and w_t, respectively, in which elements correspond to the normalized tag weights given by `Last.fm`. To overcome sparsity of tags on the track level (cf. [9]), we subsequently model each song by a joint term weight vector $w = 0.7 \cdot w_a + 0.3 \cdot w_t$, giving more weight

[2] `http://www.last.fm/api`

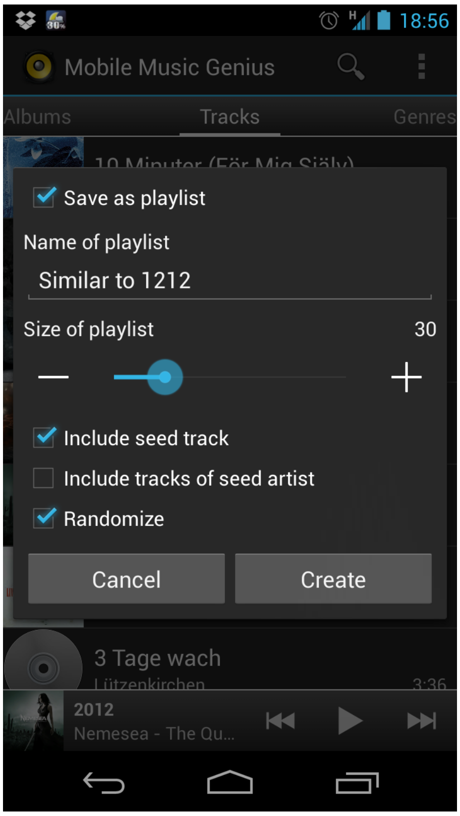

Figure 3: Seed-based playlist generation in MMG.

to artist tags. To efficiently generate seed-based playlists, MMG precomputes a similarity matrix S that contains pairwise similarities $s_{i,j}$ between all tracks in the user's collection. To this end, the inner product is used since it is computationally inexpensive, nevertheless has been shown to perform well for music retrieval tasks based on short texts [16]. If tracks are added or deleted from the music collection, S is adapted accordingly in an automated manner.

From the user's perspective, the provided interface allows to select a seed song and the user is then given the options shown in Figure 3: she can decide on the number of songs in the playlist, whether the seed track or tracks by the seed artist should be included in the playlist, and whether she wants her playlist shuffled, i.e. the nearest neighbors to the seed track randomly inserted into the playlist, instead of ordered by their similarity to the seed. The latter options give the user control over the playlist's diversity.

2.2 Playlist Creation via User Context

This method uses the current user context as well as historical user context and -interaction data to determine a seed track based on which a playlist is created using the approach presented in the former subsection. To this end, we train and re-train a classifier[3] to learn relations between user context vectors c_t and classes given by the played tracks. The user context vectors are made up of the attributes listed in Table 1.

[3]MMG makes use of a C4.5 decision tree. It is re-trained when new pairs of `<user context, track>` are created by listening events.

Given an unknown user context vector representing the current context, the trained classifier predicts its class membership probability, i.e. which tracks have been listened to in similar situations based on historical listening data. The classes (tracks) with highest probability are then used as seeds for the music context-based playlist generation. This process can be controlled by several user-adjustable parameters, e.g., time span considered for historical user context data, number of most probable tracks selected as seeds, use of feature selection to identify the most important context attributes.

2.3 Adaptive Playlist Modification

To automatically adapt the playlist depending on the user context, the listener can enable the respective option during playback. In this case, MMG continuously compares the current user context vector c_t with the previous context vector c_{t-1}, and triggers a playlist update in case $|c_t - c_{t-1}| > \rho$, where ρ is a sensitivity threshold that can be adapted by the user. If such an update is triggered, the system first compares c_t with already learned relations between user contexts and songs, based on user interaction with the player. It then inserts into the playlist, after the currently played song, tracks that were listened to in similar contexts. In order to avoid frequent repetitions of songs played in similar contexts, MMG retains a history of the last 50 recommended tracks, which is used as an exclude list. Since the classifier used to select the songs for integration into the playlist is continuously fed relations between user context and music taste, the system dynamically improves while the user is listening to music.

2.4 Implementation Aspects

The MMG player was implemented for the `Android` platform and runs on devices with `Android` 2.2 (API version 8) or higher. Contextual music and user data is stored in an `SQLite` database. MMG comes as an app as well as a widget for the device's home screen.

Given the limited amount of main memory and computing capabilities on many `Android`-powered devices, we had to make the following choices, among others: (1) restricting the dimensionality of the term feature vectors used in playlist generation to slightly over 3,000, (2) employing the simple inner product to compute term-based similarities, and (3) efficiently store the resulting similarity matrix in a database.

In addition to the adaptive playlist generation highlighted in this paper, MMG provides a variety of other functions, such as (1) automatically organizing the music collection according to artist, album, track, genre, or release year, (2) automatically retrieving album covers, performer's biographies, and descriptors from the Web, and (3) functions to manually create playlists and influence automatically generated playlists in various ways.

3. PREDICTING MUSIC PREFERENCE

To investigate how well music taste can be predicted from the user context, we built a data set by harvesting contextual data about users of the MMG player over a period of two months. In total 70 people signed up for the study, foremost students of the Johannes Kepler University Linz[4]; 42 of them actually provided context data. During this time,

[4]`www.jku.at`

we monitored about 8,000 listening events (defined by artist and track name) and the corresponding user context vectors. To prevent users from artificially adapting their listening behavior due to participating in the study, we let them use their own music collection and asked them to keep their regular music consumption behavior.

Based on the collected data, we aim at investigating to which extent we can predict a user's listening preference, given only her context. Although this is still ongoing work, first experiments with different classifiers yielded encouraging results. When predicting music artists from user context, the instance-based k-nearest neighbors classifier reached 42% accuracy, a rule learner ("JRip") 51%, and a C4.5 decision tree learner ("J48") 55%, while a simple baseline majority voter ("ZeroR") that always predicts the most frequent class only achieved 15% accuracy. This is an improvement of more than 250% over the baseline. Experiments have been conducted using the `Weka`[5] data mining software.

4. CONCLUSIONS AND FUTURE WORK

We presented the "Mobile Music Genius" (MMG) player that supports automated playlist creation and adaptation dependent on the context of the user. Static playlists are created based on similarities between collaborative tag weight profiles of artists and songs. The user can then choose to automatically and dynamically update these playlists during playback, a process that is triggered by significant changes in the user context.

Since the player is currently limited to local music collections, we foresee as part of future work the integration of streaming services, to account for their increasing popularity. We would also like to integrate methods for audio-based playlist generation in order to overcome the problems of cold start and popularity bias from which the current approach using collaborative tags may suffer.

As for predicting music listening behavior given the user context, we plan to investigate different levels of granularity. For instance, results of the corresponding classification task will likely be highly dependent on whether we predict genre, artist, or song. We will further conduct feature selection experiments to assess (1) which are the most important user context aspects and (2) to which extent those aspects vary between users.

5. ACKNOWLEDGMENTS

This research is supported by the European Union Seventh Framework Programme FP7 / 2007-2013 through the PHENICX project (no. 601166) and by the Austrian Science Funds (FWF): P22856 and P25655.

6. REFERENCES

[1] L. Baltrunas, M. Kaminskas, B. Ludwig, O. Moling, F. Ricci, K.-H. Lüke, and R. Schwaiger. InCarMusic: Context-Aware Music Recommendations in a Car. In *Proc. EC-Web*, Toulouse, France, 2011.

[2] J. T. Biehl, P. D. Adamczyk, and B. P. Bailey. DJogger: A Mobile Dynamic Music Device. In *CHI 2006: Extended Abstracts*, Montréal, Québec, Canada, 2006.

[3] D. Bogdanov and P. Herrera. How Much Metadata Do we Need in Music Recommendation? A Subjective Evaluation Using Preference Sets. In *Proc. ISMIR*, Miami, FL, USA, 2011.

[4] G. Breitschopf. Personalized, Context-Aware Music Playlist Generation on Mobile Devices. Master's thesis, Johannes Kepler University, Linz, Austria, August 2013.

[5] T. Cebrián, M. Planagumà, P. Villegas, and X. Amatriain. Music Recommendations with Temporal Context Awareness. In *Proc. RecSys*, Barcelona, Spain, 2010.

[6] S. Cunningham, S. Caulder, and V. Grout. Saturday Night or Fever? Context-Aware Music Playlists. In *Proc. Audio Mostly*, Piteå, Sweden, 2008.

[7] S. Dornbush, J. English, T. Oates, Z. Segall, and A. Joshi. XPod: A Human Activity Aware Learning Mobile Music Player. In *Proc. Workshop on Ambient Intelligence (IJCAI)*, Hyderabad, India, 2007.

[8] G. T. Elliott and B. Tomlinson. Personalsoundtrack: Context-aware playlists that adapt to user pace. In *CHI 2006: Extended Abstracts*, Montr'eal, Qu'ebec, Canada, 2006.

[9] P. Lamere. Social Tagging and Music Information Retrieval. *JNMR: Special Issue: From Genres to Tags – Music Information Retrieval in the Age of Social Tagging*, 37(2):101–114, 2008.

[10] J. S. Lee and J. C. Lee. Context Awareness by Case-Based Reasoning in a Music Recommendation System. In H. Ichikawa, W.-D. Cho, I. Satoh, and H. Youn, editors, *Ubiquitous Computing Systems*, volume 4836 of *LNCS*, pages 45–58. Springer Berlin / Heidelberg, 2007.

[11] H. Liu and J. H. M. Rauterberg. Music Playlist Recommendation Based on User Heartbeat and Music Preference. In *Proc. ICCTD*, pages 545–549, Bangkok, Thailand, 2009.

[12] R. Miotto and N. Orio. A probabilistic model to combine tags and acoustic similarity for music retrieval. *ACM TOIS*, 30(2):8:1–8:29, May 2012.

[13] B. Moens, L. van Noorden, and M. Leman. D-Jogger: Syncing Music with Walking. In *Proc. SMC*, Barcelona, Spain, 2010.

[14] O. Omojokun, M. Genovese, and C. I. Jr. Impact of User Context on Song Selection. In *Proc. ACM Multimedia*, Vancouver, BC, Canada, 2008.

[15] E. Pampalk, T. Pohle, and G. Widmer. Dynamic Playlist Generation Based on Skipping Behavior. In *Proc. ISMIR*, London, UK, 2005.

[16] M. Schedl. #nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs. *Information Retrieval*, 15:183–217, June 2012.

[17] M. Schedl, T. Pohle, P. Knees, and G. Widmer. Exploring the Music Similarity Space on the Web. *ACM TOIS*, 29(3), July 2011.

[18] Y.-C. Teng, Y.-S. Kuo, and Y.-H. Yang. A large in-situ dataset for context-aware music recommendation on smartphones. In *Proc. IEEE ICME*, San Jose, CA, USA, 2013.

[5] `www.cs.waikato.ac.nz/ml/weka`