A Daily, Activity-Aware, Mobile Music Recommender System

Xinxi Wang, David Rosenblum, Ye Wang
School of Computing, National University of Singapore
{wangxinxi,david,wangye}@comp.nus.edu.sg

ABSTRACT

Existing music recommender systems rely on collaborative filtering or content-based technologies to satisfy users' long-term music playing needs. Given the popularity of mobile music devices with rich sensing and wireless communication capabilities, we demonstrate in this demo a novel system to employ contextual information collected with mobile devices for satisfying users' short-term music playing needs. In our system, contextual information is integrated with music content analysis to offer recommendation for daily activities.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval models; H.5.5 [**Sound and Music Computing**]: [Modeling, Signal analysis, synthesis and processing]

General Terms

Application, Design

Keywords

activity classification, context awareness, mobile computing, music recommendation, sensors

1. INTRODUCTION

Users' short-term music needs are usually influenced by the users' activities, e.g., a user who is running generally will prefer loud, energizing music. In order to satisfy the users' short-term music needs better, music recommender systems need to take this context information into consideration.

Traditional music recommender systems can be classified according to three categories of recommendation methods: collaborative filtering (CF), content-based methods and hybrid methods. CF does recommendation as this: If user A and user B have similar music preferences, then songs liked by A but not yet considered by B will be recommended to B [1]. In contrast to CF, content-based systems works as follows: If user A likes song S , then songs having content (i.e., musical features) similar to S will be recommended to A [2]. Hybrid methods combine CF and content-based methods [3]. All these methods do not consider context information.

Existing context-aware music recommender systems explored many kinds of context information, such as location, time, emotional state, physiological state, running pace and weather. However, to the best of our knowledge, (1) none of the existing systems can recommend suitable music *explicitly* for *daily activities* such as working, sleeping, running, studying; (2) none of the existing systems uses mobile phones to *automatically infer* users' context in *real-time*.

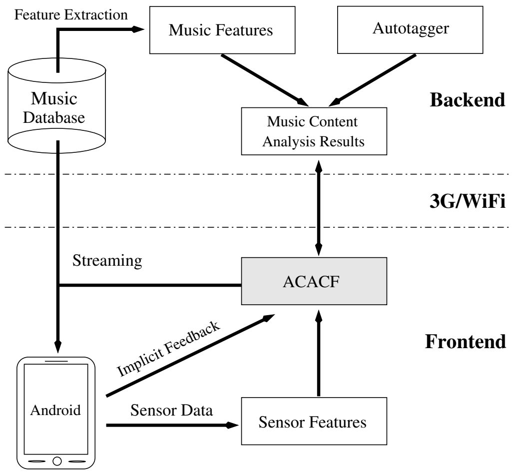


Figure 1: System Architecture

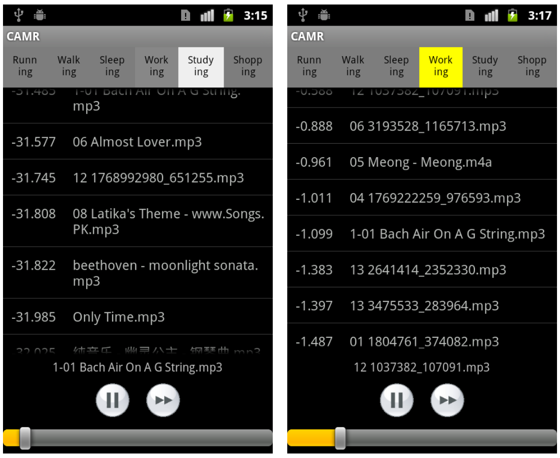
Motivated by these observations, we present a ubiquitous system built using off-the-shelf mobile phones that infers automatically a user's activity from low-level, real-time sensor data and then recommends songs matching the inferred activity based on *music content analysis* and *implicit feedback*. The architecture of the whole system is depicted in Fig. 1.

2. UNIFIED PROBABILISTIC MODEL

As the central of the system, a unified probabilistic model called Adaptive Context-Aware Content Filtering (ACACF) is built to seamlessly integrate activity inference, music content analysis and recommendation, which is depicted in Fig. 1. ACACF consists of two components: the sensor-context model and the music-context model.

Sensor-Context Model: It infers the user's activity given the feature vector extracted from the low-level sensor signal. To make it suitable for *real-time* inference on mobile phones, the model is designed to be very efficient. Model adaptation is realized like this: when the user manually specifies his current activity, the selected activity and

MM'12, October 29–November 2, 2012, Nara, Japan.
ACM 978-1-4503-1089-5/12/10.



(a) auto mode (b) manual mode

Figure 2: Context-aware mobile music recommender.

the sensor data are used to *incrementally* update the model, and thus make it more accurate.

Music-Context Model: The probability that a song suits an activity is predicted by this model. Music audio content analysis provides the initial estimation, which is used as the prior probability. The subsequent *implicit feedback* (i.e. skipped or completely listened to a song) of a particular user refines the prior by approximate Bayesian inference, which makes the recommendation *personalized* to that user.

More details are described in our full paper [4].

3. SYSTEM DETAILS

As shown in Fig. 1, the system comprises two components: the backend and the frontend.

The backend is implemented on a cluster. First, to ensure the system availability and efficiency, audio files are stored in a highly scalable and robust distributed file system[1]. Then, music feature extraction of the large scale audio files is done through the distributed computing framework MapReduce[2]. Finally, based on the music features, every song's suitable activities (i.e., music content analysis results) are predicted using Autotagger [5]. The frontend, which is a mobile application, is implemented on the off-the-shelf mobile phones. Its main interface is depicted in Fig. 2.

To stream music and retrieve the content analysis results, the frontend communicates with the backend through a wireless connection. The frontend also can run without connecting to the backend, but then songs and music content analysis results must be cached beforehand.

4. DEMONSTRATION

Activity inference: At the top of the user interface is a list of activities. Users can let the system infer his/her activity automatically, which is called the *auto mode* and is shown as Fig. 2a. The background intensity of the activity labels is adjusted according to the inferred probabilities. Users also can select one category manually, which is called the *manual mode* and is shown as Fig. 2b. When an activity is selected manually, its background becomes yellow. To switch back to auto mode from manual mode, the user needs to tap the yellow label once. Accuracy of activity classification can be checked by comparing the whitest activity label in auto mode and the actual activity of the user.

Recommendation: The list in the middle of the user interface contains the recommended songs. These songs are ranked by their posterior probabilities given the inferred activity. The accuracy of recommendation can be demonstrated by checking the top few songs of the list.

Adaptation of the sensor-context model: When the application is in manual mode, the selected activity and sensor data are used to update the sensor-context model, which makes context inference increasingly accurate. Usually, perceivable update of the model requires several minutes of data. In order to demonstrate the updating process efficiently, the learning rate needs to be set to a higher value than the normal value, and thus a few seconds of training data can significantly change the model.

Adaptation of the music-context model: Whenever the user finishes listening to a song or skips a song, the probability of the song just listened to or skipped will be updated, and all songs will be re-ranked. The songs list of the user interface will be updated immediately.

Demonstration of different activities: Our system considers six activities: running, walking, sleeping, working, studying and shopping. Running, walking, working and studying will be directly demonstrated. Sleeping needs the time to be at night and the environment to be quiet. This will be simulated by changing the time of the phone and disabling the microphone. Shopping will not be demonstrated due to the limitation of the environment.

5. ACKNOWLEDGMENTS

This work is supported by grant R-252-000-473-133 and R-252-000-428-290 from the School of Computing at the National University of Singapore.

6. REFERENCES

[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," in *CSCW*, 1994.

[2] K. Hoashi, K. Matsumoto, and N. Inoue, "Personalization of user profiles for content-based music retrieval based on relevance feedback," in *ACM MM*, 2003.

[3] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "Improving Efficiency and Scalability of Model-Based Music Recommender System Based on Incremental Training," in *ISMIR*, 2007.

[4] X. Wang, D. Rosenblum, and Y. Wang, "Context-Aware Mobile Music Recommendation for Daily Activities," in *ACM MM*, 2012.

[5] T. Bertin-Mahieux, D. Eck, F. Maillet, and P. Lamere, "Autotagger: A Model for Predicting Social Tags from Acoustic Features on Large Music Databases," *JNMR*, vol. 37, pp. 115–135, June 2008.

[1] MogileFS: https://github.com/mogilefs
[2] Hadoop: https://hadoop.apache.org