Music Recommendations with Temporal Context Awareness

Toni Cebrián, Marc Planagumà, Paulo Villegas, Xavier Amatriain

Telefónica I+D, Spain
{cebrian, mpv, paulo, xar}@tid.es

ABSTRACT

We present a system capable of recommending music playlists that take into account the temporal context of the user, i.e. they select user preferences as learned for the concrete time situation of the request.

Categories and Subject Descriptors

H.5.1 [**INFORMATION INTERFACES AND PRESENTATION**]: Multimedia Information Systems

General Terms: Algorithms, Design, Human Factors

Keywords: *Recommendations, Context, Multimedia*

1. INTRODUCTION

One recent trend in information processing and retrieval systems is the acquisition of contextual data and its incorporation to information processing [1]. Following this trend, there are approaches to improve recommendations by incorporation of contextual information in the process [2]. This can help to alleviate the problem of excessive number of hits, or to adapt better to specific needs of the user that are dependent on her current context.

Incorporation of context, however, can come at the cost of additional complexity (and computational cost) of the algorithm. In particular, observing context in a Recommender Systems converts a 2-dimensional user/item problem into an n-dimensional one through the addition of several contextual dimensions or variables.

There are three main ways to introduce context information into a Recommender System:

- *Contextual pre-filtering* uses data from the context to select the relevant data for each case. Once this is selected and filtered, regular recommending algorithms may be used.
- *Contextual post-filtering* techniques ignore contextual information and generate recommendations in a traditional way. Those recommendations are later filtered to match the current context.
- In *contextual modeling*, context information is directly fed into the model and used to derive the predictions.

Our approach belongs to the family of pre-filtering contextual recommendations. In particular, it uses a micro-profiling pre-filtering input, in which variations in the user context are used to generate a segmentation in context space and produce micro-profiles of the user, for each recurrent contextual situation. In this particular case we use context-constrained micro-profiles based on recurrent time frames. We are therefore able to generate the recommendations most appropriate for the considered context subspace.

Besides, since our approach produces a segmentation of the user profile into several contextual profiles with explainable meaning, they are easy to understand and interact with the user.

RecSys'10, September 26–30, 2010, Barcelona, Spain.
ACM 978-1-60558-906-0/10/09.

2. SYSTEM DESCRIPTION

The recommendation module is part of a bigger system for contextual processing of user info and media [3]. It implements:

- a back-end that uses a collaborative filtering approach to generate recommendations, based on user similarity generated from listening behaviour , and
- temporally-constrained user micro-profiles, used to generate the recommendations most appropriate for the considered time period.

When a user requests a recommendation, the system selects her micro-profile whose time period matches the current temporal context and obtains a recommendation list based on that profile and all the stored micro-profiles with the same time signature.

Figure 1 shows a diagram of the system. The back-end runs on a single machine and provides a REST interface for obtaining recommendations and sending notifications; the front end demonstration is implemented as a web application, making use of AJAX calls to the REST interface to obtain the recommendations. Although in the demonstration is not used, the back-end system is prepared to receive notifications from music usage, including which recommended songs get finally played, and thus update user micro-profiles accordingly.

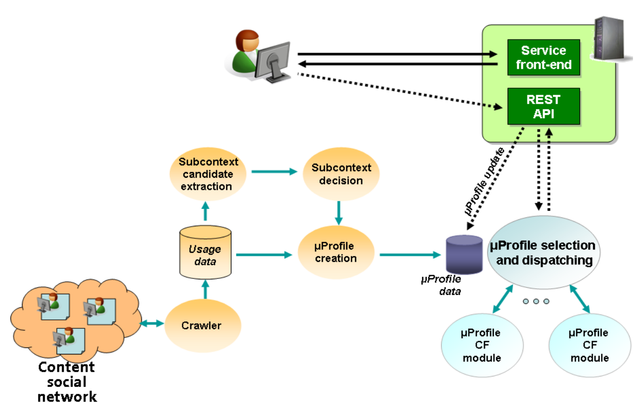

Figure 1: Overall Recommendation Architecture

3. TRAINING DATASET

The timestamped listening data was obtained from the social media site LastFM. A random sample was taken from the set of LastFM users satisfying some pre-conditions: same geographical area (Spain) and minimal amount of activity (account older than 1 year and an average minimum number of played tracks per day). The geographical limitation helps to ensure some kind of homogeneity in the space of listened artists, while the minimum activity threshold helps to reduce sparsity of data (especially important in our case since micro-profiles span only a subset of user behaviour).

The complete dataset of timestamped songs listened by those users in 2008 was then gathered off-line; it amounts to around 3.6 million items. This was the basis for micro-profile definition.

4. MICRO-PROFILE DEFINITION

By analyzing the temporal patterns in the data [4] we decided on six different micro-profiles as most discriminative, generated as the Cartesian product

(morning, afternoon, evening-night)* x *(weekday, weekend)

With those profiles defined, the dataset was split into them and a total of 7 subsets (6 for the micro-profiles and one for the global profile) was defined as input for the recommendation engines.

5. SYSTEM OPERATION

To improve live performance, recommenders work with data in memory. The timestamped dataset is dumped onto the recommender memory upon startup time, and feed two recommendation paths:

a) a time-sensitive recommender, which splits the items into the six defined micro-profiles, so that when asked for a recommendation it uses only the micro-profile relevant for the temporal context

b) a time-insensitive recommender that uses the whole dataset. This is implemented as a baseline control path

Figure 2 shows a screenshot of the demonstration interface. After logging in, the user simulates the current temporal context by selecting the relevant timeframe through a slider (shown in the lowe left). Two different playlists are then generated and shown in the interface: the time-adapted one and, as a control item, the non-contextual playlist using global user profiles (without temporal segmentation).

6. REFERENCES

[1] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in HUC '99: Proc. 1st Int. Symp. on Handheld and Ubiquitous Computing.

[2] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin, "Incorporating contextual information in recommender systems using a multidimensional approach," ACM Transactions on Information Systems, vol. 23, pp. 103-145, 2005.

[3] A. Spedalieri, A. Asensio, H. Duxans, G. Escalada, P. Villegas, "My Personal Media Entertainer: Context-Adaptive Content Recommendation and Delivery", in SAPMIA'2010 workshop at ACM MM'2010, October 2010

[4] L. Baltrunas and X. Amatriain, "Towards time-dependant recommendation based on implicit feedback,", Context-aware Recommender Systems Workshop at Recsys09, 2009.

Figure 2: Application screenshot