Automated Meta Data Generation for Personalized Music Portals

Erich Gstrein, Florian Kleedorfer, Bigitte Krenn

Smart Agents Technologies, Research Studios Austria, ARC Seibersdorf research GmbH, Vienna, Austria.
{erich.gstrein, florian.kleedorfer, brigitte.krenn}@researchstudio.at

Abstract

Providing appropriate meta information is essential for personalized e- and m-portals especially in the music domain with its huge archives and short-dated content. Unfortunately the meta data typically coming with the portal's content is not appropriate for such systems. In this paper we describe an application implemented for use in a large personalized music portal. We explain the way our system – the feature extraction engine FE^2 - generates meta information, it's architecture and how the meta data is used within the portal. Both portal and FE^2 are real world systems designed to operate on huge music archives.

1. Introduction

The distribution of music or music relevant content is currently one of the hottest topics with an enormous market potential especially in the mobile world. Due to the large scale of the data sets and the restricted usability of mobile devices, intelligent personalization systems are necessary to ensure utter satisfaction of the users [13]. Thus the more a personalization system knows about the items it recommends the better it will perform. Concerning this knowledge the content domain 'music' bears some specific traps that current commercial personalization systems must overcome, in particular these are:

Meta Data Quality: Audio content providers deliver their audio data together with only some basic information, like artist name, album name, track name, year, genre and pricing information. From a recommender's point of view – where similarity relations between items often form the basis for recommendations - this data quality is very poor, because songs are not necessarily similar if they have been created by the same artist and tracks with similar names do not necessarily sound alike.

Genres: Although a disgraced concept, it is an indispensable one to a music portal. The most serious problem is that genres are not standardized and thus are likely to be a source of dispute. For example, when it comes to music styles the AllMusicGuide offers 531, Amazon 719 and MP3.COM about 430 different genres [1].

Content volume/life-cycle: Music portals often use huge music archives (e.g., www.napster.com promises more than 1.500.000 songs) with rapidly increasing content and dynamically changing relevance of the contents (think of all the one day wonders produced by the music industry). Ensuring or creating high quality meta information is an enormous problem in this context.

Cultural dependency: The cultural background of the people interested in music plays an important role too [1], because it influences many dimensions of the selection, profiling, and recommendation components. This implies that the provided meta information also must incorporate cultural aspects.

Summing up, concerning music meta data we have to account for at least the following issues:

1. deal with the fact that content providers do not provide appropriate information for high quality recommendations
2. classify items without the availability of a sound set of classes (genres)
3. create appropriate meta information for archives that are constantly increasing in size
4. account for cultural diversity

The work presented in this paper describes an application developed to support a personalization system for a large international mobile music portal – called the *Personal Music Platform* (PMP)[13]. The PMP, currently online in Europe and Asia, offers music and music relevant products such as wallpapers, ringtones, etc. (A white labeled demo application can be visited at www.ericsson-mediasuite.com.)

The paper is organized as follows. While section 2 describes the basic concepts we build our system upon, a short overview of the architecture of FE^2 is presented in section 3. After discussing the application scenarios in section 4 the further development of FE^2 is highlighted in section 5.

2. Audio Meta Data Generation

Within recommender systems similarity is a major concept used in collaborative filtering as well as in item based filtering approaches [1, 2]. While former systems refer to similarity among users the latter focus on item similarity, especially important for the music domain where 'sounding similar' is a major selection criteria applied by end users.

But how to extract such meta information to support a personalization system? In the worst case the sources of information available to a designer of a music personalization system are:

1. a set of audio files, each coming with a title and the name of the artist
2. the web with an unforeseeable number of pages, related to some music topics, such as fan-pages, artist home pages, etc.

The field of Music Information Retrieval (MIR) has recently started to investigate respective issues, in particular the definition of similarity measures between music items. Two approaches are in focus: (1) the definition of similarity based on audio data, see for instance [3, 4, 5]; (2) the definition of similarity based on cultural aspects extracted from web pages [6, 7, 8]. In the following we will concentrate on the audio based approach.

Unfortunately the state of the art techniques for feature extraction, and similarity calculation - with similarity being defined as distance between two feature vectors - are very resource consuming. Thus the main challenge was to incorporate high quality meta data generation in a scaleable application dedicated to real world music archives.

The MIR techniques we make use of in our approach are based on MFCCs (Mel Frequency Cepstral Coefficients) which summarize timbre aspects of the music which are based on psychoacoustic considerations. For the computation of similarity between tracks the feature vectors summarizing the spectral characteristics are compared [4, 10, 11].

A serious drawback of these techniques is their time complexity. The extraction of a timbre model out of an audio file (WAVE, 22Khz, mono) takes about 30 seconds while the calculation of the distance between two vectors takes 0.05 seconds on a single machine (PC 4GHZ CPU, 1GB RAM). In the further discussion, the terms *feature vector* and *timbre model* will be used without further distinction.

Even though the extraction of the timbre model of a track takes about 30 seconds it is far less critical than the computation of the similarity relations between tracks, because of its linear behavior. Each vector has to be extracted only once and distributing the task on n machines speeds up the process at factor n. In contrast the complexity of pair-wise distance computation is $O(n^2)$ which in addition rises a storage problem (e.g. think of a complete distance matrix n x n where $n=10^6$). Therefore the optimization/reduction of distance calculations is a key factor for an application feasible for real world archives.

3. FE^2 in a Nutshell

In order to deal with the heavy workload that arises in a million tracks scenario, a distributed, easy scaleable architecture was developed that allows for parallelizing feature extraction, similarity computation and clustering jobs on multiple computers. Distribution is achieved by storing job descriptions in a database which is regularly checked for new jobs by worker nodes (Fig. 1). Distances between tracks – as well as the timbre models - are stored in a database and thus available to explorative data mining as well as to recurring tasks of content classification.

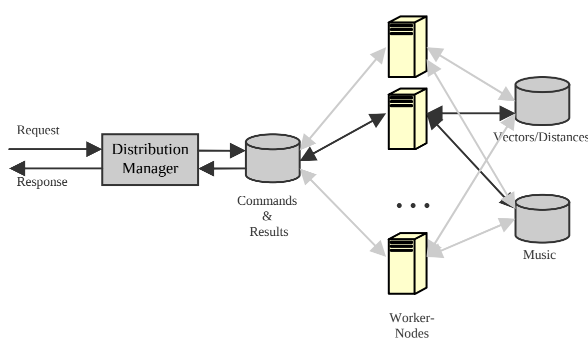

Figure 1: Distributing Requests

Continuous content growth is handled by an event-based mechanism that causes new tracks to be passed to feature extraction and subsequent classification automatically. Furthermore, the architecture is designed in an extensible way, which makes it possible to plug in additional feature extraction and model comparison mechanisms in the future.

4. Contexts of Use

In this section we describe different aspects of use of the FE^2 and its outcomes. Section 4.1 describes the most important use cases for applying similarity relations to generate appropriate meta information for improving the recommendation quality. Section 4.2 is dedicated to the process of meta data generation and highlights some administration features of the FE^2. The big picture, how the FE^2 is embedded in music portals is described in sections 4.3 and 4.4.

4.1. Applying Meta Data

The distance measures, computed by FE^2, are used for generating playlists and classes of similar sounding tracks.

Automated *playlist* generation is an important feature for a music personalization system because it supports the user in finding the most similar songs to a given track. From a technical point of view this is done by solving the *k-nearest neighbor problem*, however with the serious aggravation that the attributes of the vectors cannot be used to create common indexing structures. Instead only pair-wise distances can be used. They are computed by applying the Monte Carlo algorithm to the timbre models (implemented as Gaussian Mixture Models). For more information on distance based indexing see [9].

The classification of tracks by means of '*sounding similar*' can be seen as an alternative or complementary concept to standard genres. The classification process is performed by defining clusters $\mathbf{C_i}$ (i=1..n) via a set of prototypes $\mathbf{P_{ij}}$ (timbre models) and assigning a class to each track of the audio archive. A track $\mathbf{T}$ is affiliated to a cluster $\mathbf{C_i}$ if the distance $\mathbf{DIST(T, C_i)}$ is smaller than any other $\mathbf{DIST(T, C_j)}$. The distance $\mathbf{DIST(T, C_i)}$ is computed by defining the minimal distance between the timbre model of $\mathbf{T}$ and one of the prototypes $\mathbf{P_{ij}}$ of $\mathbf{C_i}$. In the course of the PMP project we pursue two different approaches: manual cluster generation and semi automated cluster generation.

In the case of handpicked clusters (*manual cluster generation*), the prototypes are defined by a human expert and the resulting classification process is used to support the content administrator. In PMP this approach is mainly used for defining mood clusters – such as 'feeling blue', 'feeling excited', etc. – containing music that best matches the given mood.

Semi automated cluster generation is performed by applying clustering algorithms (e.g. k-means clustering) on a sample set of the audio archive followed by a tuning /cleaning process by the content administrator. Within the PMP project this approach is mainly used to define a more intuitive genre concept based on sound similarity.

4.2. Supporting the Content Administrator

Being confronted with hundreds of thousands of tracks and with a time consuming classification process an appropriate tool supported modus operandi is essential for an industrial application. Apart from the technical aspects of scalability and performance the support of the content administrator is one of the most important aspects of the FE^2. The core features are:

- different sample and test sets can be pulled out of the archive
- several sample sets can be classified in parallel
- the affiliation of tracks to clusters is represented graphically
- the consequences of using a track as prototype of a cluster are displayed on-line
- cluster metrics and tests provide information about the quality of the clusters

4.3. FE^2 in the Context of the Music Portal

In the context of the PMP the feature extraction engine is used twofold:

1. as a batch process for classifying items against a defined set of prototypes, triggered by the content feed process of the portal
2. as an administration tool used by a content administrator to define and refine classification schemes

The information flow between PMP and the FE^2 is bidirectional. In a first step the meta data information calculated by the FE^2 is imported into the PMP portal to boost the personalization system. In a second step user feedback concerning the quality of the meta data collected by the personalization system is employed to refine the meta data generation process.

For the time being the following kinds of feedback are incorporated in the meta data generation process:

- the affiliation of tracks to predefined, mood specific music genres like 'feeling blue', 'excited', etc. by the user. These tracks are used to refine/define the set of the specific cluster prototypes
- users ratings on elements of recommendation lists are utilized to tune the 'playlist' and the 'similar artist' generation process

4.4. 3rd Party Applications

Companies like *Gracenote* (www.gracenote.com), *All Music Guide* (www.amg.com) or *Hifind* (www.hifind.com) are creating high-quality meta-information on the basis of the human expert knowledge. The opponents of high-quality hand crafted content are up-to-dateness, focus on mainstream and of course the costs as a killer argument for small or medium size portals.

The application area of automatic meta data generation software like FE^2 is therefore not only limited to the improvement of specific portals (like PMP) but also can generally improve/support an editorial approach as illustrated in section 4.2.

5. Future Direction

Beside the ongoing improvements of the audio based approach the capability of our FE^2 will be extended to other sources of information. Promising approaches have recently been presented that try to exploit lyrics [12] or analyze the content of websites [6]. These approaches are complementary to the audio analysis and are particularly suited to capture cultural information. Concerning the FE^2 we currently explore the applicability of 'artist similarity' – based on features extracted from websites – and 'track similarity' based on lyrics. Furthermore, the applicability of visualized similarity relations (e.g. visualization of clusters) for improving navigation will be investigated.

6. References

[1] A. Uitdenbogerd, R.v. Schyndl. A Review of Factors Affecting Music Recommender Systems. Dep. of Computer Science, Melbourne, Australia.

[2] Herlocker, J., L., and Konstan, J., A., Terveen, L., G., and Riedl, J., T., (2004). Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information systems, Vol. 22 No. 1, January 2004, Pages 5 – 53

[3] Pampalk, E, Dixon, S., and Widmer, G. (2003). On the Evaluation of Perceptual Similarity Measures for Music. In Proceedings of the 6th International Conference on Digital Audio Effects (DAFx-03), London.

[4] Aucouturier, J.J and Pachet, F. (2002). Music Similarity Measures: What's the Use? In Proceedings of the International Conference on Music Information Retrieval (ISMIR'02), Paris, France.

[5] Aucouturier, J.-J. and Pachet, F. (2004). Improving Timbre Similarity: How High isthe Sky? Journal of Negative Results in Speech and Audio Sciences 1(1).charge of

[6] Knees, P., Pampalk, E., and Widmer, G. (2004). Artist Classification with Web-based Data. In Proceedings of the 5th International Conference on Music Information Retrieval(ISMIR'04), Barcelona, Spain, October 10-14, 2004.

[7] Whitman, B. and Lawrence, S. (2002). Inferring Descriptions and Similarity for Music from Community Metadata. In Proceedings of the 2002 International Computer Music Conference, pp 591-598. 16-21 September 2002, Göteborg, Sweden.

[8] Baumann, S. and Hummel, O. (2003). Using Cultural Metadata for Artist Recommendation. In Proceedings of the International Conference on Web Delivery of Music (WedelMusic), Leeds, UK.

[9] Bozkaya, T. and Ozsoyoglu M., (1999). Indexing Large Metric Spaces for Similarity Search Queries. ACM Transactions on Database Systems, Vol. 24, No. 3, September 1999, Pages 361–404.

[10] Foote, J.T. (1997). Content-based Retrieval of Music and Audio. In Proceedings of the SPIE Multimedia Storage and Archiving Systems II, 1997, vol. 3229.

[11] Logan, B. and Salomon, A. (2001). A Music Similarity Function Based on Signal Analysis.

[12] Logan, B., Kositsky, A., and Moreno, P. (2004). Semantic Analysis of Song Lyrics. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME).

[13] Gstrein, E., et al. (2005). Adaptive Personalization: A Multi-Dimensional Approach to Boosting a Large Scale Mobile Music Portal. In Fifth Open Workshop on MUSICNETWORK: Integration of Music in Multimedia Applications, Vienna, Austria.