

Browsing Music by Usage Context

Valentin Laube, Christian Moewes, and Sebastian Stober

Faculty of Computer Science
Otto-von-Guericke-University Magdeburg, D-39106 Magdeburg, Germany
valentin.laube@st.ovgu.de
{cmoewes, stober}@cs.uni-magdeburg.de

Abstract. This paper aims to motivate and demonstrate how widely available environmental data can be exploited to allow organization, structuring and exploration of music collections by personal listening contexts. We describe a logging plug-in for music players that automatically records data about the listening context and discuss possible extensions for more sophisticated context logging. Based on data collected in a small user experiment, we show how data mining techniques can be applied to reveal common usage patterns. Further, a prototype user interface based on elastic lists for browsing by listening context is presented.

1 Introduction

Indeed several studies indicate that there exist usage patterns that people consciously or unconsciously use when they access music collections or describe music: In a user study [14] that analyzed organization and access techniques for personal music collections, several “idiosyncratic genres” could be identified that users tend to use to classify and organize their music. These idiosyncratic genres comply largely with the usage context. Typical examples could be “music for driving (and keeping me awake)”, “music for programming” or “music to relax in the evening after a long working day”. Further, an analysis of requests at the answering service “Google Answers”¹ in the category “music” [2] revealed that such descriptions were also used in this public setting. In a larger survey [16] on search strategies for public music retrieval systems, more than 40% stated that they would query or browse by usage context if this would be supported by the system. Strong correlations between genre, artist, album and the usage context also became apparent in a more recent study [12].

Even for the construction of an objective taxonomy of 378 music genres [18], features referring to consumer context (“audience location”) and usage (“danceability”) were important criteria. Such meta-data is already exploited in a commercial application [10] to select music for a desired atmosphere in hotels, restaurants and cafes. However, the respective properties need to be assigned manually by experts and if necessary can only be adapted by hand.

It would be very desirable to have at least a semi-automatic context assignment from automatically retrievable or measurable data. According to the

¹ <http://answers.google.com/answers/>

definition by Dey [8], any information that can be used to characterise the situation of a person, place or object of consideration makes up its context. He differentiates four types of primary context: location, identity, time, and activity [9]. In the music information retrieval domain, there exists already a variety of systems that capture time, (user) identity and location, e.g. the Audioscrobbler² plug-in from last.fm³. However, this information is rarely used to describe a usage context and to our knowledge it has so far not been used for personalized access to music collections. Recalling the phenomenon of idiosyncratic genres, this yields a high potential for supporting an individual user in maintaining and using his personal music collection.

In this paper, Section 2 gives an overview of related music retrieval systems that incorporate information about the usage context. Subsequently, we describe our logging plug-in for music players that automatically records data about the listening context. Using data collected by the logger plug-in in a small user experiment, we demonstrate in Section 4 how data mining techniques can be applied to reveal common usage patterns. Further, we present a basic prototype user interface in Section 5 that allows browsing the music collection by listening context. In Section 6, further extensions for context logging are discussed. Finally, Section 7 concludes our work.

2 Related Work

In the field of ubiquitous computing several recommender systems for music have been described that use easily measurable environmental data to differentiate between listening contexts:

The M^3 music recommender system described in [15] uses a two-step case-based reasoning approach for context-aware recommendation. First, information about season, month, day of the week, weather and temperature is used to infer whether the user wants to listen to some music. This decision is made through case-based reasoning on the user's listening history. If music is likely to be desired, a second case-based reasoning step infers, whether the music should be slow, fast or may have any tempo. Here, the tempo is estimated from the genre tag, assuming that songs belonging to the genres "Ballad" and "R&B" are slow whereas songs belonging to "Rock/Metal" and "Dance" are fast.

In [19], a context-aware music recommendation system is described that apart from weather (temperature, humidity, current weather and forecast) and time (season and time of day) data also incorporates the ambient noise level recorded by a microphone and the illuminance measured by a sensor. The continuous data is discretized to fuzzy membership vectors with respect to predefined fuzzy sets. The resulting data is processed by a bayesian network that infers the current context. Explicitly created user-profiles are then used to recommend songs with regard to the context.

² <http://www.audioscrobbler.net/>

³ <http://last.fm/>

Finally, a music recommender system for the smart office is proposed in [11]. The system uses basic content-based classifiers to assign the available songs to distinct genre and mood classes. Songs are recommended that comply with the genres specified in the user profile and match the user's current mood. The user's mood is predicted by a naive bayesian classifier that takes into account the users location, the time of day, which other people are in a room with him, the weather outside and his stock portfolio.

In contrast to these approaches that all target at a recommendation scenario, we aim to exploit available environmental information to allow for browsing of music collections by personalized listening contexts. This involves a deeper analysis of the context data by means of machine learning and data mining techniques.

Apart from collecting environmental information, listening context information can also comprise direct information about the user's current condition. For instance, the adaptive system for playlist generation called PAPA (Physiology and Purpose-Aware Automatic Playlist Generation) [17] as well as the already commercially available BODiBEAT music player⁴ uses sensors that measure certain bio-signals (such as the pulse) of the user as immediate feedback for the music currently played. This information is then used to learn which characteristics of music have a certain effect on the user. Based on this continuously adapting model playlists for certain purposes can be created. Alternatively, it could be used to derive listening contexts.

Though directly measuring bio-signals is highly interesting, it requires special sensors and most importantly demands a high tolerance of the user. In contrast to this, environmental information can mostly be gathered at low costs either from internet resources (e.g. weather information) or with onboard hardware such as a built-in microphone, a webcam or gyroscopic sensor and illuminance sensors. Further, it can be measured without distracting the user. That is why we focus in our work only on environmental data.

3 Data Acquisition

For logging the context information together with the played songs, a plug-in for the foobar2000, Winamp and iTunes music player was developed. Whenever a song is played, the plug-in records its ID3 metadata together with a time stamp and the "end reason", i.e. whether the song played till the end, was skipped or the player was closed before the song ended. Further, information about the local weather conditions is gathered from online services. The location of the user is estimated by resolving the IP address of the computer. If the computer is offline, data is gathered once it is re-connected with the internet. The recorded data is cached in a local SQLite database⁵ and transferred in constant intervals via HTTP to a central server that collects the data for analysis.

⁴ <http://www.yamaha.com/bodibeat/>

⁵ <http://www.sqlite.org/>

Table 1. Discretized values used for data mining. (weather quality is discrete)

attribute	(discretized) values
time of day	morning (5-8), forenoon (8-11), noon (11-14), afternoon (14-17), evening (17-20), night (20-23), late night (23-5)
weather quality w	sunny, mostly sunny, partly sunny, clear, mostly clear, partly cloudy, mostly cloudy, cloudy, overcast, light fog, mist, light snow, snow shower (sleet) snow, drizzle, light rains shower, light rain, rain shower, rain
temperature ϑ	$<0^{\circ}\text{C}$, $0..5^{\circ}\text{C}$, $5..10^{\circ}\text{C}$, $10..15^{\circ}\text{C}$, $15..20^{\circ}\text{C}$, $\geq 20^{\circ}\text{C}$
pressure p	$<900\text{ hPa}$, $900..1000\text{ hPa}$, $1000..1050\text{ hPa}$, $\geq 1050\text{ hPa}$
pressure change Δp	neg. big ($<-10\text{ hPa}$), neg. medium ($-10..-5\text{ hPa}$), neg. small ($-5..-2\text{ hPa}$), zero ($-2..2\text{ hPa}$), pos. small ($2..5\text{ hPa}$), pos. medium ($5..10\text{ hPa}$), pos. big ($>10\text{ hPa}$)

In a small test experiment with 8 participants, 15325 played songs were logged between February and April 2008. The data comprises the following 14 dimensions: user id, artist, title, album, genre, date (from ID3) end reason \mathcal{Y} , weekday, time of day t , weather quality w , temperature ϑ , humidity, air pressure p , pressure change (during the last four hours) Δp . However, because of initial problems with the logger, a large number of the records is not complete. Moreover, the data is biased towards bad weather and about half of the data has been contributed by a single user. This is important to keep in mind when assessing the data mining results presented in the following section.

4 Data Mining

In order to find commonly used patterns or useful information from the acquired music data, we applied several data mining techniques from the data analysis platform Information Miner⁶. We focussed on learning the dependency between the weather conditions \mathcal{X} and the reason why a user ended a song $\mathcal{Y} = \{\textit{finished}, \textit{skipped}, \textit{quitting}\}$. Formally, this problem can be described as finding a function $f : \mathcal{X} \mapsto \mathcal{Y}$.

First, the data was projected to a subset of attributes, i.e., \mathcal{Y} and $\mathcal{X} = \{t, w, \vartheta, p, \Delta p\}$. In the second preprocessing step all records containing missing values were removed. After this step, only 2064 records remained for further analysis. As a final step before the data mining we discretized all continuous variables as shown in Table 1. Using these attributes, we applied several techniques from the Information Miner toolkit.

Figure 1 shows an induced graphical network structure [5] that was generated by applying the K2 metric [6] to all variables \mathcal{X} and \mathcal{Y} . Edges indicate interdependencies of attributes. Not surprisingly, the interdependencies shown in the network match with common sense knowledge. For instance, the time of day has an impact on the temperature, the air pressure and the weather condition (cond). Analyzing the impact of the time of the day and the weather condition on the end reason, several rules can be generated from the model. They are shown as circles in Figure 2, plotted by lift and recall and colored with respect to the end reason. The most interesting rules are close to the top right corner.

⁶ <http://fuzzy.cs.uni-magdeburg.de/wiki/pmwiki.php?n=Forschung.InformationMiner2>

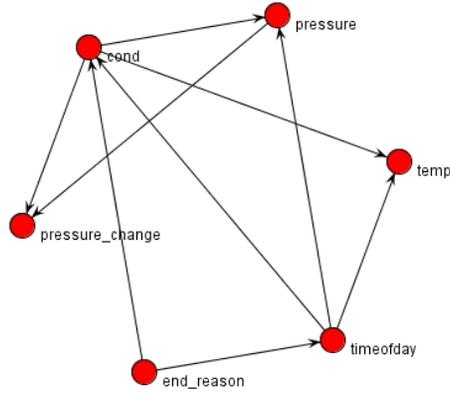


Fig. 1. The most probable graphical model given the data. Edge directions can be ignored in this case as only interdependencies are of interest. Note that the applied K2 metric tries to maximize the probability of a directed acyclic graph given a database of sample cases.

Figure 3 shows a decision tree [20] learned on the same attributes as the graphical model with \mathcal{Y} set as the class variable. Values of the same attribute were grouped into single nodes wherever possible to reduce the complexity of the tree [3]. The resulting tree was pruned to a maximum height of 4. Several selection measures were tested in order to find reasonable tree structures of which the sum of weighted differences showed the most promising results. In the induced tree, every path from the root to a leaf node corresponds to a rule that can be directly derived. For instance, if there is snow shower, snow, light rain shower or mist in the afternoon or evening, then there is a $256/298 = 86\%$ chance that the user will finish the song.

In order to apply frequent pattern mining [4] to our problem, we used the Apriori algorithm [1] for finding maximum item sets. The only difference between the previous tests and this one is the extended set of attributes. Here, the weekday was considered additionally. The identified frequent item sets with their relative support are listed in Table 2. From these item sets, rules can easily be constructed by putting all attribute values in a table row except for \mathcal{Y} into the antecedent (precondition). The consequent is simply determined by \mathcal{Y} .

Finally, naïve and full Bayesian classifiers [5] were trained on both the discretized and the raw data. However, the results were not satisfactory and the induced rules were also harder to interpret. Therefore, detailed results are omitted here. Yet, we want to mention that Bayesian learning will possibly come up with useful information if there is more data with less missing values.

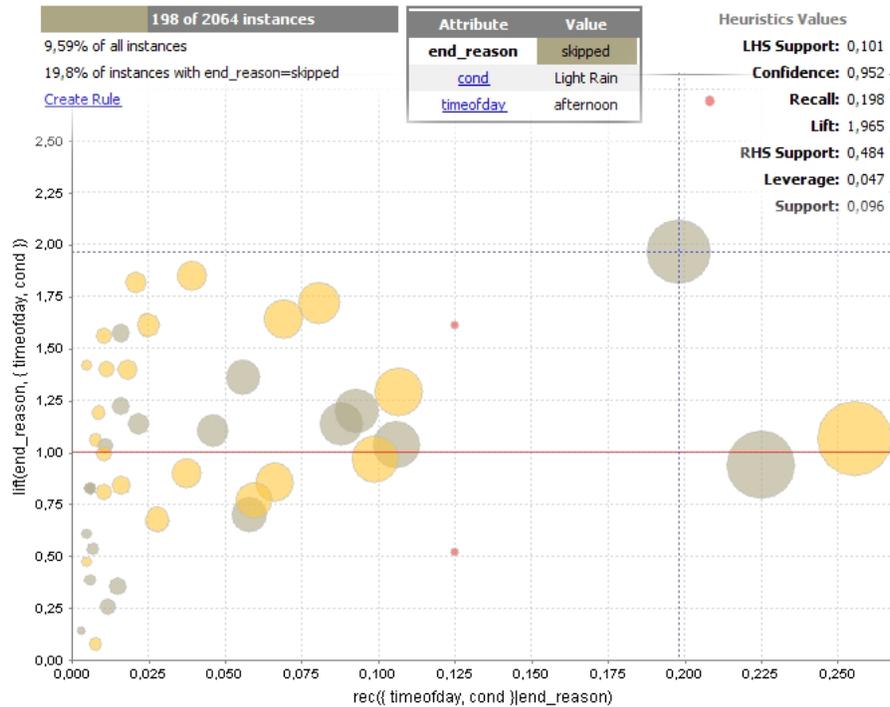


Fig. 2. Some found association rules for the end reason plotted by their recall and lift. The colors represent the different end reasons, i.e., yellow corresponds finished, grey to skipped and red to quitting. The selected rule (blue cross hairs) is as follows: In case of light rain in the afternoon, then there is a 10% chance of skipping a song. Note that the surface of each circle is direct proportional to its rule's relative number of instances.

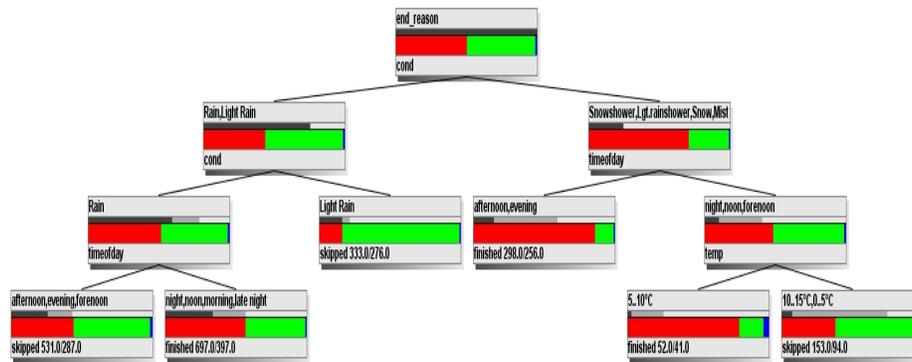


Fig. 3. Decision tree for the end reason. Tree nodes consist of three rows: selected attribute values (top), value distribution of the end reason (middle with red=finished, green=skipped and blue=quit) and the final decision and its accuracy (bottom).

Table 2. Induced maximum item sets ordered by descending relative support. Only item sets containing an item from \mathcal{Y} and a minimum support of 10% are shown.

\mathcal{Y}	t	w	ϑ ($^{\circ}\text{C}$)	p (hPa)	Δp	weekday	rel. supp.
skip		rain	5..10	1000..1050	zero		16.0%
finished		rain	5..10	1000..1050	zero		15.2%
skip		light rain					13.4%
finished						Saturday	12.3%
skip	afternoon		0..5				11.8%
skip	afternoon					Saturday	11.3%
skip			0..5			Saturday	11.0%
skip	night	rain			zero		10.9%
finished	night	rain		1000..1050	zero		10.8%
skip	night		5..10	1000..1050	zero		10.7%
finished		light rain shower					10.6%
finished					zero	Monday	10.6%
skip			10..15				10.4%
finished			0..5		zero		10.3%
finished	evening			1000..1050	zero		10.1%
finished			10..15	1000..1050	zero		10.1%

5 Browser Prototype

For a first prototype user interface that allows browsing by listening context, we adopted the elastic list technique [21] that was developed for browsing multifaceted data structures.⁷ This approach enhances traditional facet browsing interfaces such as presented in [7] for music collections that allow a user to explore a data set by filtering available metadata information. In the scope of this work, we use the available context metadata as facets, i.e. user, time of day, day of week, weather condition, temperature, air pressure and air pressure change (during the last 4 hours). As only discrete features are supported, we use the discretized version of the features. Further, the logged ID3 metadata (artist, title, album and genre) can also be used naturally as facets. A screenshot of the interface is shown in Figure 4

Additionally to facet browser filtering, elastic lists visualize relative proportions of values by size. For instance, the sizes of the blocks referring to the days of the week in the respective facet column reflect the distribution of the number of played songs on these days. Selecting some value of a facet as filter will update the proportions of the blocks in all other facet columns, now reflecting the distribution of the facet values under the given filter constraints.

Further, elastic lists visualize unusualness by brightness. We slightly adopted this approach to visualize negative and positive deviations. If, for instance, the end reason “skipped” is selected and for some value of a facet the number of skipped songs is significantly higher or lower than the expected average value, then the respective block is colored red or green respectively. Brighter colors indicate a stronger deviation from the expected value. For instance, in Figure 4 the selection of user #8 shows that in this context, significantly more songs are finished than usual.

⁷ An online demo of the original elastic lists user interface can be found at http://well-formed-data.net/experiments/elastic_lists/

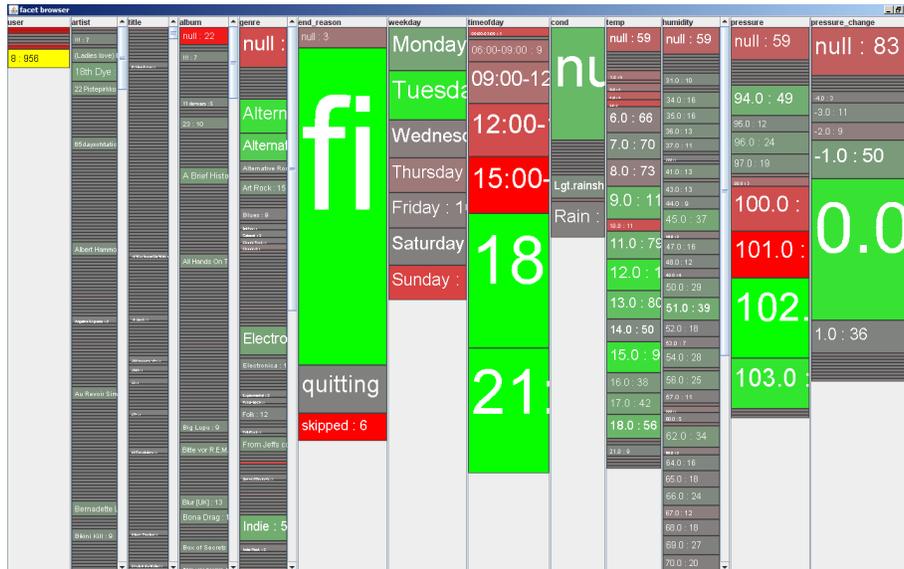


Fig. 4. Context browser prototype. User #8 has been selected (yellow). He mostly lets the songs play till the end which is significantly more probable than for the average user (green). He usually listens to music in the evening and at night (green) and less often than other during the day (red).

6 Possible Extensions for Context Logging

As the context data currently recorded is very basic, there is a variety of possibilities for extension. Looking at the related systems discussed in Section 2 the most obvious extensions would be to add sensor information about the environment, e.g. about the illuminance and the background noise level. Indeed, recently some notebooks are equipped with illuminance sensors to adapt the display brightness. The information from these sensors could be used as a context. It is however questionable whether this context information is really helpful and not e.g. already sufficiently covered by the time of day. The background noise can easily be recorded by a built-in microphone as available in most devices that are capable of playing music. Further, many notebooks and some recent mobile phones are equipped with gyro sensors that could also provide additional sensor information. Regarding the location context, especially for mobile devices, more sophisticated logging could also involve gathering GPS coordinates. As many mobile phones and handhelds nowadays have a GPS receiver, even this would not require additional hardware.

Recalling the four primary contexts mentioned by Dey [9], i.e. location, identity, time, and activity, the latter context is currently not covered. However, this one is especially important because it is closely related to the idiosyncratic genres identified in [14]. Moreover, knowledge about the user's current activity

might enable a more sophisticated modelling of the listening context with regard to the listening modes defined in [13].

A very basic method would be to simply ask the user, what he is currently doing. As this requires a user action without a directly recognizable benefit there is hardly any motivation for the user to cooperate. An alternative would be to try to automatically guess the activity. If the user uses multiple devices for playing music (and all devices have a logging mechanism), the device itself and its location context can be a rough indicator for the activity. Music played on the office PC will most likely be a background for office work whereas music played on the car radio is listened to while driving. Another option would be to exploit available sensor data mentioned above. GPS data could indicate locations linked with specific activities or travel. Combined with data from gyro sensor, e.g. certain sport activities could be recognized. This, however, lies far beyond the scope of this work.

A more promising option is to exploit the background noise information. Obviously, this information is not immediately available as the recorded signal will most likely contain the currently played music unless headphones are used. Thus, as a preprocessing step, it would be necessary to filter the music from the signal. Luckily, the music signal is known and there exist already some good method for removing a known signal from a mix. The resulting background sound could be classified according to some generic categories such as silence, people talking, nature sounds, traffic sounds or party. As many notebooks and mobile phones have also a built-in webcam, this approach could be extended even further by adding video information to improve classification accuracy. However, the more sophisticated this analysis gets, the closer it resembles an audiovisual surveillance scenario. This might put the user acceptance of the system at risk.

Assuming the person is using a computer, further possibilities arise. Detecting whether and how much the mouse and keyboard are used in a sliding time window yields evidence about the user's activity. For instance, low keyboard and mouse activity may indicate reading or browsing whereas high keyboard activity may refer to writing a text or programming. It might not even be necessary to derive such higher level activity description. The low level information might be already sufficient to distinguish activity contexts. More sophisticated computer activity logging would be to analyze, which applications are currently running and which application has the focus. However, this again comes close to surveillance.

To sum up, there are many possibilities for extended context logging but clearly privacy is the most important issue. So the question is not, what is technically possible but how much information about his activities a user is willing to share. As more sophisticated method come closer to surveillance, the user must be fully informed about the extend of the logged data and in full control of whether he wants this data to be logged or not. In addition, it needs to be shown, that this additional information is indeed helpful, i.e. the user has a benefit from providing this information.

We plan to investigate the usefulness of computer activity logging and extended background noise logging with a generic classification because they do not have special requirements in terms of hardware and appear to provide quite promising context information. These extension will be added as optional plug-ins for the logger that a user may enable or disable according to his preferences.

7 Conclusions

We have presented a basic logging plug-in for music players that records context information about the user, the time and the local weather. Further possibilities for context extensions were motivated and discussed. Based on data collected in a small user experiment, we show how data mining techniques can be applied to reveal common usage patterns. However, the data is very biased, especially towards bad weather due to the recording period. For more significant results, data collected over a longer time period is needed. We have further presented a prototype context browser with a visualization that enables users to understand how context metadata values are correlated with each other, which is often interesting information itself. For the near future, we plan to extend the logging capabilities of the music player plug-in and to combine the user specific context data with content features within an application for personalized organization and structuring of music archives.

8 Acknowledgements

We would like to thank all participants of the user study and Matthias Steinbrecher for providing support during the data analysis. This work is supported by the German Research Foundation (DFG) and the German National Merit Foundation.

References

1. R. Agrawal, T. Imieliński, and A. Swami. Mining Association Rules between Sets of Items in Large Databases. In *Proc. Conf. on Management of Data*, pages 207–216, New York, NY, USA, 1993. ACM Press.
2. D. Bainbridge, S. J. Cunningham, and J. S. Downie. How people describe their music information needs: A grounded theory analysis of music queries. In *Proc. of ISMIR'03*, 2003.
3. C. Borgelt. A Decision Tree Plug-In for DataEngine. In *Proc. 6th Europ. Congress on Intelligent Techniques and Soft Computing (EUFIT'98)*, volume 2, pages 1299–1303, Aachen, Germany, 1998. Verlag Mainz.
4. C. Borgelt. Efficient Implementations of Apriori and Eclat. In *1st Workshop of Frequent Item Set Mining Implementations FIMI 2003*, Melbourne, FL, USA, 2003.
5. C. Borgelt and R. Kruse. *Graphical Models - Methods for Data Analysis and Mining*. J. Wiley and Sons, Chichester, United Kingdom, 2002.
6. G. Cooper and E. Herskovits. A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning*, 9:309–347, 1992.

7. R. Dachsel and M. Frisch. Mambo: a facet-based zoomable music browser. In *MUM '07: Proc. of the 6th Intl. Conf. on Mobile and ubiquitous multimedia*, pages 110–117, New York, NY, USA, 2007. ACM.
8. A. K. Dey. Understanding and using context. *Personal Ubiquitous Comput.*, 5(1):4–7, 2001.
9. A. K. Dey and G. D. Abowd. Towards a better understanding of context and context-awareness. In *Computer Human Interaction (CHI) 2000 Workshop on the What, Who, Where, When, Why and How of Context-Awareness*, April 2000.
10. S. Govaerts, N. Corthaut, and E. Duval. Moody tunes: the rockanango project. In *Proc. of ISMIR'06*, 2006.
11. D. Guan, Q. Li, S. Lee, and Y. Lee. A context-aware music recommendation agent in smart office. *Fuzzy Systems and Knowledge Discovery*, pages 1201–1204, 2006.
12. X. Hu, J. S. Downie, and A. Ehmann. Exploiting recommended usage metadata: Exploratory analyses. In *Proc. of ISMIR'06*, 2006.
13. D. Huron. Listening styles and listening strategies. Society for Music Theory 2002 Conference, November 2002.
14. S. Jones, S. J. Cunningham, and M. Jones. Organizing digital music for use: an examination of personal music collections. In *Proc. of ISMIR'04*, 2004.
15. J. Lee and J. Lee. Music for my mood: A music recommendation system based on context reasoning. *Smart Sensing and Context*, pages 190–203, 2006.
16. J. H. Lee and J. S. Downie. Survey of music information needs, uses, and seeking behaviours: Preliminary findings. In *Proc. of ISMIR'04*, 2004.
17. N. Oliver and L. Kreger-Stickles. PAPA: Physiology and purpose-aware automatic playlist generation. In *Proc. of ISMIR'06*, 2006.
18. F. Pachet and D. Cazaly. A taxonomy of musical genres. In *Proc. of Content-Based Multimedia Information Access (RIAO)*, Paris, France, 2000.
19. H.-S. Park, J.-O. Yoo, and S.-B. Cho. A context-aware music recommendation system using fuzzy bayesian networks with utility theory. *Fuzzy Systems and Knowledge Discovery*, pages 970–979, 2006.
20. J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1:81–106, 1986.
21. M. Stefaner and B. Muller. Elastic lists for facet browsers. *dexa*, 0:217–221, 2007.