Interactive Visualization for Music Rediscovery and Serendipity

Ricardo Dias   Joana Pinto
INESC-ID, Instituto Superior Técnico, Universidade de Lisboa
Portugal
{*ricardo.dias, joanadiaspinto*}*@tecnico.ulisboa.pt*

Manuel J. Fonseca
Faculdade de Ciências, Universidade de Lisboa
Portugal
mjf@di.fc.ul.pt

Although personal tastes may change over time, overall people still enjoy music they have not listen to for some time or not very often. However, most solutions for browsing music collections do not focus on showing or suggesting these songs to create serendipitous rediscoveries. Instead they promote most recently played songs as an entry point for browsing and playing music. This way, users are constantly listening to the same music, causing the least heard to become forgotten. In this paper, we present BACH, an interactive visualization and exploration tool for personal music collections that uses the listening history of the user to influence the songs suggested and the way they are presented to him/her. Our goal is to help users rediscover their music collection for different periods of the day through the perspective of their listening history. Experimental results revealed that users understood and enjoyed our solution and that they were able to rediscover their collections by listening to songs heard some time ago.

Exploration, Interactive Visualization, Music Discovery, Listening Histories

1. INTRODUCTION

Over the last few years, we have assisted to a huge rising and growth of music streaming services, such as, *Spotify* (8), making the listening experience ubiquitous, as people can listen to music at anytime and everywhere. A subproduct of the listening process are listening histories, which are a sequence of songs listened over time by a user. These are easy to collect and provide valuable details about users, since they encode users' musical tastes.

Different solutions have been developed over the years both for visualizing and browsing music collections and listening histories. Typical solutions for browsing music collections use metadata and content-based features from audio, web-information and social features, to represent collections applying different visualization techniques such as treemaps(5), SOMs (11) or graphs. On the other hand, techniques for visualizing listening histories are more concerned in allowing users to interactively browse their listening histories, while leading them to identify listening trends and habits (1). Although, listening histories have been used to improve recommendation algorithms, there is still a scarce integration of these into techniques for browsing music collections, not taking advantage of these to improve and complement the visualizations.

Figure 1: *BACH User Interface.*

In this paper, we present BACH, an interactive visualization and exploration solution for browsing personal music collections (see Figure 1). It offers an hourly-based grid visualization split in two areas. The outermost area shows the recently played songs (*Fresh songs*) while the innermost presents a set of songs not played for some time (*Frozen songs*). We aim to provide to users a tool for the rediscovery of their music collection for different hours of the day through the perspective of their listening history, by suggesting frozen songs similar to those recently played for that hour. To increase transparency and to engage users in browsing tasks, we offer brushing and highlighting mechanisms to visualize and explain suggestions.

Experimental evaluation with users showed that our solution is easy to understand and to use and that the visual explanation of the suggestions is a good added value. Moreover, our main goal was achieved, since users accepted our suggestions of frozen songs, rediscovering their collections.

The contributions of this research are i) An interactive visualization technique for music collections, integrating metadata and listening histories; ii) An algorithm for recommending frozen songs based on the users' current musical tastes; iii) A visual representation of the suggestions mechanism to promote transparency and trustiness.

2. RELATED WORK

To achieve our goal of promoting the rediscovery of songs people have not listen for a long time or were listen less frequently, we analyzed related work on techniques for browsing and visualizing music collections, and techniques for representing listening histories. The analysis of these browsing techniques allowed us to understand their benefits and problems in conveying information to users.

Several visualization techniques have been employed to browse music collections, from maps, to graphs, clusters and other less standard techniques.

In *Islands of Music* (11), the authors used a 2D islands and sea metaphor for representing songs based on acoustic similarity, with water representing the absence of songs, and the islands a concentration of songs. Lillie developed MusicBox (9), a technique that maps the music collection into a 2D space, applying PCA to a combination of contextual and content-based features from songs. Dias et al. (5) applied content-based similarity for browsing personal collections using treemaps both for visualization and filtering. Other less standard visualizations include the MusicRainbow (10), a circular rainbow visualization, or the technique proposed by Hopmann et al. (7), an interface for navigating digital collections based on a one-dimensional analog control and a data visualization based on old analog radios. Musicovery[1] is a web-radio that plays songs according to our mood, allowing users to select a mood from a point in 2D space. Liveplasma is an approach for exploring similarities between artists that uses data from Amazon to create a similarity graph, connecting all the artists.

Over the past few years, some solutions and visualizations have been developed to specifically visualize listening history information.

Byron and Wattenberg, back in 2008, developed a kind of stacked graph, called *StreamGraph* (4), to visualize trends in personal music listening. Some fan-created static visualizations were also created, ranging from timelines displaying the number of logged songs, or arcs diagrams displaying how frequently listening habits change (or how mainstream specific music tastes are); to stacked-graphs, like the *Last.fm Explorer* (12) or the LastGraph[2], where interactivity and support for exploration within restrictive perspectives was added.

Baur and Butz (2), using a force-directed node-and-link diagram and some variations, developed three track-based visualizations for this kind of data and proposed an automatic *playlist* generation mechanism. Later, the authors developed LastHistory (1), a solution that used context information from personal calendars and photographs. In (3), Baur et al. proposed a visualization to explore and compare the differences / similarities between two talks about music. The visualization compares two listening histories in a timeline technique using the similarity between two songs and the relevance of the song to the talk. Dias et al. developed an approach for browsing listening histories by combining a rich-featured timeline-based visualization with an interactive filtering mechanism, with the goal of helping users to identify trends and habits (6). Their experimental evaluation with users revealed that users were able to infer about their main life events and listening changes, and also promote new behaviors.

Although there are various solutions for each area, so far none of them tried to combine listening history methods with visualization and browsing techniques to enrich the exploration of music collections.

3. SOLUTION DESIGN AND RATIONALE

With our solution we want to allow users to rediscover their music collections for the different hours of the day, starting from their listening histories. To achieve this, we suggest frozen songs that have not been heard for some time, but are still within the musical tastes of the user for that period of the day, by using artist similarity.

We expect our solution to answer the following questions:
Exploration: Does the hourly-based visualization help users browse their collections? The brushing and highlighting techniques promote both browsing and recommendation explanation? Do users rediscover frozen songs just by visualizing their collection?
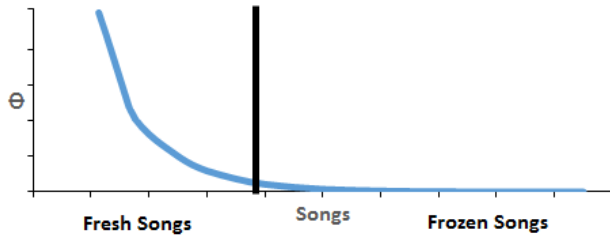
Figure 2: *Division of the music collection.*

Suggestion: Do users understand the reasons behind the suggested songs? What do they feel about those songs? Are they surprised with the proposed songs? Do they select fresh or frozen songs for their playlists?

3.1. Data Model, Organization and Structure

Recently conducted research revealed that, although personal tastes may change over time, the choices for each period of the day remain consistent through time (6). Moreover, users still enjoy listening to less recently played songs, representing them, sometimes, a fresh and serendipitous rediscover. Based on these, our solution performs an hourly division of the collection, creating a set of songs for each hour of the day. Finally, we assume that all songs from the user's collection were listened at least once (to be part of the listening history).

Guided by the goal of leading people to rediscover frozen songs that still are in their musical tastes, we divide each hour dataset in two groups: fresh songs and frozen songs. To perform this separation we devised a factor Φ (see Equation 1), which combines memory retention (the number of times a song was played) and freshness (the time since the last reproduction).

$$\Phi = \frac{Current\ Date - Last\ Reproduction\ Date}{Number\ of\ Reproductions} \quad (1)$$

We compute the Φ factor for each music in the hour dataset and we sort them by decreasing order as depicted in Figure 2. We consider the first third as the group of fresh songs and the remaining two thirds as frozen songs. The latter group will be included in the pool of songs (composed by the frozen songs from every hour dataset) from which our solution will select the songs to suggest, based on the similarity with the artists in the former group. This division allows us to keep the listening context for each hour, and suggest similar songs to those listened at that time period, promoting music rediscovery. We chose this division based on preliminary experiments, to maintain a good ratio between fresh and frozen

songs. These two datasets created for each hour are the basis data model used by our visualization method.

3.2. Visualization and Interaction

BACH user interface is composed of 4 distinct areas, as illustrated in Figure 1: 1) most recently played songs; 2) suggested songs; 3) controls; and 4) playlist. As unit of representation of the music collection we chose the song, which is represented by its album art. On top of this, we depict four buttons, as shown in Figure 4: 1) add to playlist; 2) information details; 3) similarity highlighting; and 4) audio preview.

The most recently played songs area (Fig.1-1) is located on the external rectangle of the interface. Songs are grouped by artist and ordered alphabetically by his/her name, from top to bottom, left to right. We organize songs by artists, because the similarity between songs is determined using artist-similarity. The suggested songs area (Fig.1-2) is located at the inner rectangle, in the center of the display, and uses the same representation as area 1.

The control area (Fig.1-3) is located at the bottom of the interface, consists of 24 buttons (one for each hour), and two sets of buttons to navigate on the pages of the list of fresh songs and the list of suggested songs. Whenever the user selects an hour, the visualization changes to display the most fresh songs for that hour (which provides the context for the suggestions) and the corresponding suggested songs (similar to those in the context).

The playlist area (Fig.1-4) is a window that becomes visible when the user moves the cursor over the right side of the interface. Users can add and remove songs from the playlist, as well as re-order them.

Users can browse and explore their music collections by using the buttons in the control area, and also by using the similarity button located in the lower left corner of the songs artwork (see Figure 4). This button allows users to highlight the similarity between songs and to get details about the recommendation process for a particular song, regardless of the area where it is. Thus, we can see why a particular suggestion was made and also what suggestions were made because of a particular fresh song (see Figure 3). This *highlighting* mechanism was developed to engage users in the browsing tasks, by providing a greater interactivity, but essentially as a way to explain the similarity between songs and consequently the relationship between fresh songs and suggested songs. By clicking on this button, only that song and the related ones stay visible, while the others fade out. This way,
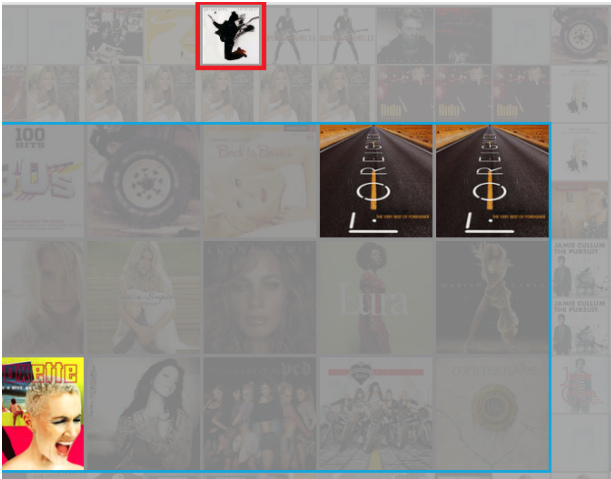
Figure 3: *Songs suggested by a recently listened song (red border). This evidences the highlighting mechanism.*
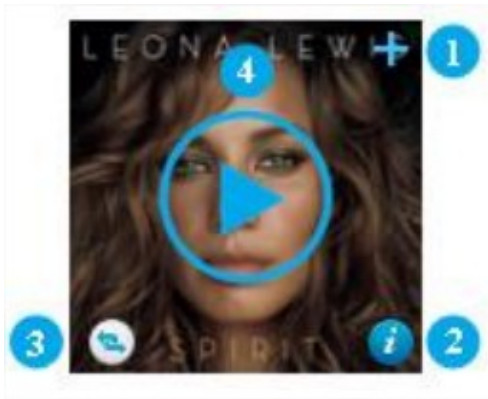

Figure 4: *Song visualization and interaction buttons.*

the user can freely explore and get more information about those songs (*brushing*).

4. EVALUATION

The main goal of our solution is to allow users to rediscover their music collection, by suggesting frozen songs that are similar to those that users have been listening recently. To check whether our objectives have been met, we conducted an experimental evaluation with users to evaluate the usability of our tool and the changes in the Φ factor of the music collection, using both objective and subjective measures. The objective measures were the Φ factor of the songs over time and the number of frozen songs added to the playlist. The subjective measure was the users satisfaction with the tool.

4.1. Goals and Tasks

The main goals of these tests were to see if the average value of the Φ factor decreases over time and if our solution helps increase the number of songs with a lower value of Φ. This way, we can

conclude that users have been recently listening to frozen songs (those suggested by our solution). We also wanted to check if users liked the new perspective of their collections provided by our solution, where (similar) frozen songs are showed together with recently listened music and if it allowed them to rediscover their music collection.

To collect data, we asked users to perform a set of tasks on our solution, consisting in creating playlists. Users created four playlist for a specific hour of the day, using songs from any set of music (fresh and/or frozen). The playlists were created one at a time, with the goal of simulating the passing of time. The idea was that each playlist would correspond to a day, and due to that the Φ factor of the songs will change, since one more day had passed. For the tests we select the hour of the day with more songs, which was different for each user.

4.2. Participants

We recruited 11 users (all males), with ages within the interval from 18 to 60 years old, with the majority of them (72.7%) in the interval from 18-30. All users listen to music on a daily basis, were registered on the Last.fm service and scrobbled most of their listening activity. They had a personal listening history with an average age of two years containing between 4,000 and 5,000 songs. The average number of songs for the hour selected for the tests was between 260 and 330 songs. Users were recruited personally and through email, Facebook and Last.fm. Each user performed the evaluation using their own collection.

4.3. Procedure

Each user performed the tests on his/her own without any member from the development team supervising the tests. To that end, we provided a usability test script to each participant, containing a short explanation of the tool, a link to the online application, the description of the tasks to perform and a link to the satisfaction questionnaire.

Before performing the requested tasks, we allowed users to freely explore the application and all its functions. Only after that learning period users performed the requested tasks. After creating the four playlists users were requested to fill in a satisfaction questionnaire composed by a set of questions, most of which were rated using a 5-point Likert scale, ranging from Completely Disagree to Completely Agree. We also had some open ended questions to collect comments and suggestions from users. With this questionnaire we wanted to evaluate the overall usability and functionality of our solution, and also to check the understanding of some of the

design options we took during the creation of the visualization of the application.

4.4. Results

In general, all users included suggested (frozen) songs in the playlists showing that they were rediscovering their collection. Moreover, users understood the visualization mechanism being able to explore their collections and liked the way songs were presented in two sets and by hours of the day, allowing them to understand their listening patterns.

4.4.1. Music Rediscovery

Figure 5 shows the normalized Φ factor values for the overall music collections for a (simulated) sequence of five days. It presents the median, maximum, minimum and the first and third quartiles of the Φ factor, for a specific hour of the day. As we can see the median decreases over time revealing that the number of songs with a smaller Φ factor increased. This means that songs which were not heard for some time were played recently. Furthermore, the decrease of the first quartile, tending to zero, supports this conclusion, meaning that we have more songs recently played. This is also supported by the histogram in Figure 6, where we can see that the number of (fresh) songs with the smallest values of Φ factor (between 0 and 0.1) increases 86%, while (frozen) songs with the biggest values (between 0.9 and 1) decreases 10%. The decrease is smaller than the increase, because our solution suggests frozen songs from all hours of the day and not only from the selected hour under analysis.
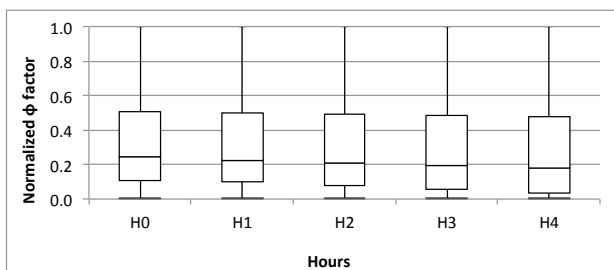

Figure 5: *Normalized Φ factor for five days.*

On all created playlists users included more than 50% (between 50 and 60%) songs from the suggested list (frozen songs). This reveals that the suggested songs were still interesting for the users, allowing them to rediscover their collection.

4.4.2. User Satisfaction

Our satisfaction questionnaire was divided into three parts. One served to characterize users about their listening habits. The second part was used to check if the main goals of our solution were achieved, namely the rediscovering of the collection and the new provided perspective of the music
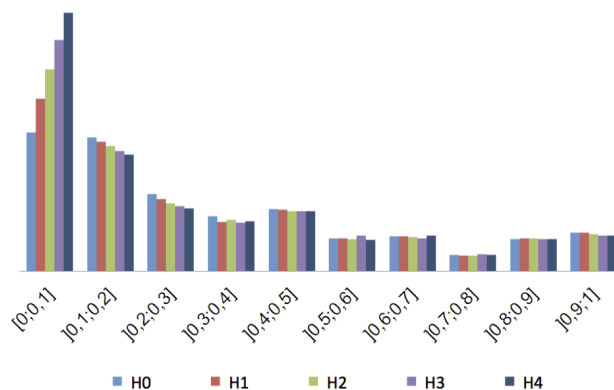

Figure 6: *Histogram evolution for five days.*

collection. Finally, the last part tried to validate some of the design decisions that we took for the visualization, such as the separation between fresh and suggested songs, the mechanism to visually explain the suggested songs and the overall navigation.

From the satisfaction questionnaire we found that users *Agree* with all the statements in the questionnaire such as: 1) the tool helped them to figure out that the suggested songs were not heard for a long time; 2) it helped them to rediscover songs they liked but that they did not heard for a time; 3) the tool was useful for them to rediscover their music collection; 4) it helped them to get a new perspective of their collection; 5) the suggested songs still were in their preferences; 6) they were able to understand the visualization mechanism that shows how frozen songs were suggested; 7) they considered useful the visualization mechanism that shows how frozen songs were suggested; 8) the distinction between fresh and frozen songs was easy to understand; 9) the distribution of songs by hours of the day was useful for searching their collections; 10) they would like to continue using the solution to rediscover their music collections. Additionally, eight out of eleven users (72.7%) said that they would recommend the tool to other people.

During the experimental evaluation we also collected some comments and suggestions from users. We noticed that users liked the organization of their collections based on the listening history and the hour of the day; the separation between fresh songs and suggested songs; the emphasis given to the suggested songs; the fact that the suggested songs were aligned with their musical tastes; and that they were able to rediscover songs that they still like but that they do not remember they have. Some users said: *"I was able to rediscover my collection", "I liked the division by hours of the day and the mechanism that allows me to understand what music leads to the*

suggested songs", "The division by hour of the day allowed me to understand better my listing patterns".

About the limitations and suggestions, users consider that just showing the art of the album is not enough to provide a quick information about the song. They suggest using a short textual description with the artist and song name. Other suggestions to improve the solution were to add some filtering mechanism, the suggestion of new songs not in the collection, or a method to group songs by mood, genre or bpms.

4.5. Discussion

From the results we can conclude that our solution achieved its main goal, since (frozen) songs listened some time ago are heard again. This way users are listening not only to the most recent songs, but also to old ones that they liked in the past. About the usability of the application, users considered it easy to understand and to use, and clearly understood the visualization used to represent the two sets of songs. Users liked the visualization based on the listening history, which provided them a new perspective of their collections. Users considered important the mechanism to identify and represent the songs that originated the suggested songs.

In summary, we can say that our solution leads users to listen to songs that they did not heard for some time, and that it is able to suggest songs similar to those that users like to listen at that hour of the day.

5. CONCLUSIONS

In this paper we presented BACH, a tool for browsing personal music collections through the perspective of past listening history. It presents to users the most recently played songs together with suggested songs that were not listened for some time and that are similar to the former. Our approach divides the collection in two sets by computing the Φ factor of the songs. A brushing and highlighting mechanism complements the interactive visualization by visually explaining the source of the suggestions. Regarding future work, we plan to explore similarity between songs (instead of artist) to promote trustiness, and include more textual information about the songs, as users considered it essential for a quick identification of the song.

6. ACKNOWLEDGMENTS

REFERENCES

[1] D. Baur, F. Seiffert, M. Sedlmair, and S. Boring. The streams of our lives: Visualizing listening histories in context. *IEEE TVCG*, 2010.

[2] Dominikus Baur and Andreas Butz. Pulling strings from a tangle: visualizing a personal music listening history. IUI, 2009.

[3] Dominikus Baur, Andreas Butz, and Sheelagh Carpendale. arcs.fm - a backdrop visualization for music talk. In *EuroVis*, 2012.

[4] L. Byron and M. Wattenberg. Stacked graphs geometry & aesthetics. *IEEE TVCG*, 2008.

[5] Ricardo Dias and Manuel J Fonseca. Muvis: an application for interactive exploration of large music collections. In *Proc. of ACM MM*, 2010.

[6] Ricardo Dias, Manuel J. Fonseca, and Daniel Gonalves. Interactive exploration of music listening histories. In *Proc. of AVI*, 2012.

[7] Mathieu Hopmann, Mario Gutierrez, Frédéric Vexo, and Daniel Thalmann. Vintage radio interface: Analog control for digital collections. CHI EA, 2012.

[8] Jin Ha Lee and Nichole Maiman Waterman. Understanding user requirements for music information services. In *ISMIR*, 2012.

[9] Anita Shen Lillie. Musicbox: Navigating the space of your music. Master's thesis, MIT, September 2008.

[10] Elias Pampalk and Masataka Goto. Musicrainbow: A new user interface to discover artists using audio-based similarity and web-based labeling. In *ISMIR*, 2006.

[11] Elias Pampalk, Andreas Rauber, and Dieter Merkl. Content-based organization and visualization of music archives. In *Proc. of ACM MM*, 2002.

[12] Maxwell A. Pretzlav. Last.fm explorer: An interactive visualization for hierarchical time-series data. *University of Californica, Berkeley*, 2008.