Culture Unbound
Journal of Current Cultural Research

More of the Same – On Spotify Radio

By Pelle Snickars

Abstract

Spotify Radio allows users to find new music within Spotify's vast back-catalogue, offering a potential infinite avenue of discovery. Nevertheless, the radio service has also been disliked and accused of playing the same artists over and over. We decided to set up an experiment with the purpose to explore the possible limitations found within "infinite archives" of music streaming services. Our hypothesis was that Spotify Radio appears to consist of an infinite series of songs. It claims to be personalised and never-ending, yet music seems to be delivered in limited loop patterns. What would such loop patterns look like? Are Spotify Radio's music loops finite or infinite? How many tracks (or steps) does a normal loop consist of? To answer these research questions, at Umeå University's digital humanities hub, Humlab, we set up an intervention using 160 bot listeners. Our bots were all Spotify Free users. They literally had no track record and were programmed to listen to different Swedish music from the 1970s. All bots were to document all subsequent tracks played in the radio loop and (inter)act within the Spotify Web client as an obedient bot listener, a liker, a disliker, and a skipper. The article describes different research strategies when dealing with proprietary data. Foremost, however, it empirically recounts the radio looping interventions set up at Humlab. Essentially, the article suggests a set of methodologies for performing humanist inquiry on big data and black-boxed media services that increasingly provide key delivery mechanisms for cultural materials. Spotify serves as a case in point, yet principally any other platform or service could be studied in similar ways. Using bots as research informants can be deployed within a range of different digital scholarship, so this article appeals not only to media or software studies scholars, but also to digitally inclined cultural studies such as the digital humanities.

Keywords: Spotify Radio, digital methods, music looping, bot intervention, reverse engineering

Snickars, Pelle: "More of the Same—On Spotify Radio", *Culture Unbound*, Volume 9, issue 2, 2017: 184–211. Published by Linköping University Electronic Press: http://www.cultureunbound.ep.liu.se

Introduction – "Same Artists Over and Over"

Sometime during Winter 2014 someone posted a question on the Q&A site Quora: "Why does my Spotify radio sound so repetitive? I feel I am getting a few artists repeated". At the time, Spotify Radio had been around for more than two years, but Quora users were disappointed. "The radio functionality in Spotify is very crude", the Finnish 'info junkie' Heikki Hietala replied to some frustrated listeners. Maybe Spotify will "come up with something soon, but as for now Spotify Radio is very annoying". Apparently, Hietala had the same experience of repeated songs being played, and instead recommended the music streaming service Pandora. According to Hietala, the latter had way more successfully "chopped the music up into tiny pieces of metadata [delivering] a truly mesmerising radio function due to the vast amount of information they have on the music" (Hietala 2014).

Quora is a question-and-answer site where users post questions, which are subsequently answered, edited, and organized by the community of users on the same site. Queries on Quora often address tech, which is not surprising since the company was co-founded by two former Facebook employees and is based in Mountain View, California (Google headquarters). Quora also seems to be a site frequented by tech employees themselves, which makes it particularly interesting from a research perspective. Former Tech Lead at Spotify, Erik Bernhardsson, has published almost 30 posts, some with references to discussions on Spotify Radio.

A couple of months after Hietala's post, another objection in the same vein re-appeared on Quora. In fact, almost identical questions around the inferior functionality of Spotify Radio kept being posted: "How do I get Spotify to stop playing the same few songs for every artist?"; "How do I teach a Spotify radio station to play a wider array of songs?"; "Is the Spotify streaming radio . . . purposefully terrible with the intention of trying to get people to upgrade?" (Quora 2016). Within our ongoing research project on Spotify, we have discussed similar issues around the poor performance of the Spotify Radio algorithm. Naturally, such assumptions reveal a normative claim that the radio algorithm should produce apt recommendations. To answer at least some of these issues, we decided to set up an experiment that would explore Spotify Radio. Essentially, we wanted to uncover why we (usually) didn't like the songs the radio algorithms suggested we should like. But given normative assumptions about the ways in which Spotify Radio ought to work, the research question was also broader, hinting at the ways in which algorithmic music discovery today features and promotes some artists and simply ignores others. A software driven cultural analyses of music delivery mechanisms could potentially reveal the algorithmic flaws that regulate music recommendations in disfavour of, say, more diverse listening, making less room for emerging musicians or neglected genres (with economic ramifications). Ultimately, an investigation of Spotify Radio would also stress the possible limitations and restraints found within "infinite archives" of music streaming services.

Our hypothesis was that many streaming services' radio functions appear to consist of an infinite series of songs. For commercial reasons, Spotify Radio claims to be both personalised and never-ending, yet music seems to be delivered in limited loop patterns. If our hypotheses held true, what would such loop patterns look like? Are Spotify Radio's music loops finite or endless (given that its algorithm(s) can choose between 30 million songs)? How many tracks (or steps) does a typical loop consist of? Importantly, how is the size of a music loop on Spotify Radio affected by user interaction in the form of likes, dislikes, and skips? Does, for example, *n* amounts of likes expand the music loop in terms of novel songs and artists?

This article describes different research strategies and digital methods when dealing with proprietary data as well as the background and the establishment of the radio functionality at streaming services like Spotify. It briefly recounts, for example, what is known about Spotify Radio's music discovery engine. Foremost, however, the article empirically recounts, discusses, and analyses the radio looping interventions set up at the digital humanities hub, Humlab (Umeå University, Sweden). Essentially, the article suggests a set of methodologies for performing humanist inquiry on (mid-size) big data and black-boxed media services that today increasingly serve as key delivery mechanisms for cultural materials. Spotify serves as a case in point, yet principally any other platform or service could be studied in similar ways.

Proprietary Data & Research Strategies

One of the users on Quora concerned with the second-rate quality of Spotify Radio was Web designer Bas Leijder Havenstroom: "Why does my Spotify Radio play the same artists over and over for me?" In a re-entry in the thread that followed, he specified what he was puzzled about:

> I re-asked this one because this frustrates me . . . Even if I start a radio station based on a playlist with many, many artists, I find that some (specific) artists keep coming back. I have the feeling that this all has to do with commercial reasons. I believe record labels pay Spotify to have their artists to show up in radio stations and random functions more often. (Leyder, Havenstroom 2015)

The major problem in doing contemporary research on streaming music is that claims like the one made by Leijder Havenstroom cannot be tested. The specifics of Spotify's algorithms are proprietary, and statistics around listener data are a corporate secret. Some rudimentary data are available via the various Spotify interfaces, ranging from the number of streams for popular songs to artist followers and listeners per month. At the site yearinmusic.spotify.com data with the most popular tracks, albums, or playlists (divided, for example, geographically or by time of year) are also available. If a user logs in, personal statistics are displayed in a similar manner. In addition, Spotify occasionally releases listener data and statistics, usually in various tie-ins with magazines or newspapers to gain public attention. Basically, the same strategy (regarding the lack of access to user data) is deployed by other music streaming services. In short, not much data is available.

Consequently, academically there are gaps and lack of knowledge about the ways in which algorithmic music discovery takes place (Kjus 2016). Because of the "lack of transparency in how recommendations and 'discoveries' are presented", as Jeremy Wade Morris and Devon Powers have stated, it is not clear, for example, how promotional messages for artist are featured. At Spotify, an advanced promotional feedback loop today mixes user activity with interface design, and the line between the service as "a distribution outlet" and as a "promotional intermediary" becomes completely blurred (Morris & Powers 2015). Hinting at the metaphorical associations between the streaming service Pandora and Pandora's box in Greek mythology, Paul Allen Anderson has even claimed that the former connotes "a black box of friendly mysteries" (2015). From a media research perspective, it is simply hard to tell how "music recommendation works – and doesn't work", to quote a blog post from Brian Whitman, CTO of the Echo Nest and Principal Scientist at Spotify. Admittedly, computational knowledge and access to data does not automatically lead to profound insights regarding music discovery or the broader relation between culture and tech. Still, people such as Whitman (or Bernhardsson) on the inside do know more (or used to know), but for media researchers it is hard to tell (Whitman 2012). Gaining access to the inside becomes important, or as a former Spotify intern and researcher, Sander Dieleman, stated: "At Spotify, I have access to a larger dataset of songs, and a bunch of different latent factor representations obtained from various collaborative filtering models" (2014). In addition, given that personalisation algorithms alter user experience through interactions with the system, Nick Seaver notes that "it is very difficult, if not impossible for a lone researcher to abandon the subject position of 'user' and get an unfiltered perspective" (2014b).

As a result, research undertaken around algorithmic music discovery has been done from a strict technical perspective within the field of computer science (Shao et al. 2009). Computationally-oriented studies have been made regarding, for example, how music recommendations based on artist novelty and similarity work (Lin et al. 2014) or implementing recommendation system based on user behaviour. These studies tend to focus on either applying self-developed algorithmic systems or evolve around mathematical models with, for example, the use of "Gaussian distributions to evaluate each possible genre for the next track" (Yajie & Ogihara 2011). Due to the difficulty (or even impossibility) in obtaining valid data from streaming services, media studies' perspectives on algorithmic discovery, on the other hand, have tended to favour hermeneutic and critical explorations – i.e., traditional humanistic readings of algorithmic music recommendations (Allen Anderson 2015; Modell 2015; Morris & Powers 2015) or interview-based examinations, sometimes done from an ethnographic or anthropological perspective. Nick Seaver, for example, is currently finishing a long-term anthropological study of developers of music recommender systems (2016). As an anthropologist, he is sceptical about working with digital methods: "While reverse engineering might be a useful strategy for figuring out how an existing technology works, it is less useful for telling us how it came to work that way" (2016). The risk of using digital methods (as reverse engineering), according to him, is that technical details "hidden behind the curtain" become the sole purpose of research. Unlocking corporate secrets, such as how algorithmic discovery works, is not only about technology: "Not everything worth knowing has been actively hidden" (Seaver 2014a).

Seaver's argument is worth considering. At the same time, he advocates a methodological blind alley, sticking to a traditional and long-established interview-based or participant observation methodology (in his case with software developers). Naturally, such methods can offer valuable insights into the thinking that goes into building algorithms. Then again, within our research project we are mainly interested in using new digital tools to understand the politics of algorithms from a completely different angle. Unlike Seaver, our research project engages in reverse engineering Spotify's algorithms, aggregation procedures, metadata, and valuation strategies, breaking into the hidden infrastructures of digital music distribution, to study its underlying norms and structures. One point of departure is that Spotify resembles a black boxed service, metaphorically as well as practically (at least from an academic media studies perspective). Another point of departure is that Spotify does not, to put it bluntly, share any data. Lack of access to data today confronts media scholars, (digital) humanists, and social science researchers working with social media studies. "Twitter determines what data are available and how data can be accessed through [their] API", David Gunnarsson Lorentzen, for example, states in his thesis, *Following Tweets Around* (2016). How Twitter and other similar platforms and services (like Spotify) provide access to data influences how researchers conduct their work: "The central problem [is] that researchers do not know what relevant data are not collected" (Gunnarsson Lorentzen 2016: ii).

Then again, even if Twitter data are biased, research has flourished and benefitted enormously from the ease of access to the relatively open data the service provides, at least in comparison to strictly confined music streaming services. Since Spotify user data are not available, the data must be acquired and compiled through other means to perform research, for example, by deploying bots as research informants. As stated, in our experiments we wanted to explore and investigate how Spotify Radio generally functions. What we set out to do in our interventions was to use hundreds of bots to compile user data and essentially reverse engineer what a radio loop at Spoitfy looks like. As a research strategy, reverse engineering starts with the final and implemented product, in our case Spotify Radio within the streaming service desktop client, and tries to take it apart "seeking clues as to why it was put together in the way it was and how it fits into an overall architecture" (Gehl 2014:10). As an attempt to reveal the procedures of culture and technology at work, reverse engineering can be linked to various forms of hacking practices. Within media studies, reverse engineering has been used both by academic scholars (Friesinger & Herwig 2014) as well as by tech journalist wanting to understand and analyse, for example, how Netflix's sorting algorithms, vocabulary, and grammar work (Madrigal 2014). In cases where the code remains black-boxed, Rob Kitchin has stated that "a researcher interested in the algorithm at the heart of its workings is left with the option of trying to reverse engineer the compiled software". Referring to Seaver's critique, Kitchin has endorsed the use of bots as a research strategy when dealing with proprietary code and inaccessible data. One solution for enhancing clarity is "to employ bots, which posing as users, can more systematically engage with a system, running dummy data and interactions" (Kitchin 2016). This is exactly what we have been doing in our radio looping interventions.

A Brief Spotify Radio History

The Spotify Radio slogan states that the service "lets you sit back and listen to music you love. The more you personalize the stations to match your tastes the better they get" (Spotify Radio 2016). Users of Spotify can start a radio station based on artist, song, playlist, album, or even genre. For users, Spotify Radio is a "lean back experience", yet with the ability to tune recommendations with thumbs up (like), thumbs down (dislike), or by skipping a song. It is one of many discovery mechanisms or packages associated with music recommendation systems at Spotify, including Discover, Related Artist, Genres, and Moods, Discover Weekly, and lately, Release Radar. In short, the radio functionality allows people (via various algorithms) to find new music within the vast back-catalogue of Spotify, offering a potential infinite avenue of discovery. It is important to stress, however, that the concept of radio has served as the point of departure for the recommendation business within streaming music services. Hence it is essential to study how different radio functionalities have developed, since the radio metaphor for recommendations seems to be diminishing, gradually being replaced by other and more sophisticated modes of machinic suggestions.

According to Noa Resare, Spotify's Free Software Advocate, when a user starts the radio functionality, the Spotify Web client makes a request to "a specialized radio service which holds radio selections for a large subset of all our material" and "[t]he radio suggestions are built using analysis of previous playback data" (2013). Despite the significant amount of data that goes into radio recommendations, for many the feature has been a disappointment. The impression is confirmed if one reads threads and conversations on the Spotify community Web, where the radio functionality is repeatedly criticised. The function, in fact, aroused disappointment right from the start: "Better radio algorithms . . . there are too many repetitions", one user stated already after the launch in 2012 (Lehwark). The critique has remained, and a search using Google autocomplete even suggests the amusing "Spotify radio always [plays the] same song" (if one starts a query for "Spotify radio algorithm").

By and large, the general concept of a personal digital radio station began around 2000. At the time, the idea was introduced to create a separate and individualised radio station for each user depending on personal preferences. In the U.S., within the Music Genome Project, a mathematical understanding of music was developed, which used some 450 attributes to describe and dissect music. Experts tagged songs with different characteristics (genre, instruments, tonality, etc.). Using this information, algorithms organised and bundled music in specified ways. Commercially, the Music Genome Project formed the backend and core technology of Pandora Radio, which after 2000 became the first music streaming and automated recommendation service. Its popularity grew, reaching an estimated 70 million monthly users in 2013. Due to music rights restrictions, however, Pandora Radio is only available in the U.S., Australia, and New Zealand.

For several years, Pandora Radio was widely recognised as the best music recommendation service, and Spotify (all likely) tried to replicate its personalised music offerings. Nonetheless, Spotify seemed to have lacked technological expertise, and as a result started co-operating with the music intelligence company The Echo Nest. In a blog post in December 2011, it was announced that the latter company would now power the new Spotify Radio:

> Spotify has over ten million users . . . But there's one thing Spotify didn't have until now: artist and song radio stations. Thanks to Spotify's deal with The Echo Nest, users of the popular music service can now create streaming radio stations based on any artist or song on Spotify. The stations are generated by The Echo Nest's Playlist API, and are

> available both to free and premium Spotify subscribers. (The Echo Nest 2011)

This co-operation developed, and in April 2012 Spotify began updating its desktop software with several new features, including a Pandora-like radio station: "Spotify to Take On Pandora With Radio service" (Hachman 2012). Arguably, the Spotify Free model, with ad-supported unlimited streaming access, was especially important in gradually establishing the service in the U.S. An online radio offering, it was proclaimed, "would advance Spotify's strategy of attracting users with free, ad-supported services who can be converted later into paying subscribers" (Fixmer 2012). At the time, The Echo Nest did not exclusively power Spotify's radio recommendations. Since its API was open, competitors like Rdio and Deezer were also using it. In March 2014, however, Spotify acquired The Echo Nest, a deal that was said to strengthen its music discovery expertise: "The acquisition supports Spotify's strategy to grow global music consumption and overall revenue back to the music industry by building the best user experience and music discovery engine for millions of global fans" (The Echo Nest 2014).

According to some press accounts, during the last years Spotify has put The Echo Nest employees in charge of its most important discovery products (Popper 2015). Nevertheless, the music discovery engine at Spotify, technologically sustained by The Echo Nest, remains obscure. Basically, the same uncertainty and unpredictability makes it difficult to research the different algorithms regulating music recommendations on Spotify Radio. In addition, recommendation algorithms vary and the music discovery engine has naturally, like most computational systems, been altered, improved, and updated since its initial release (Chandra 2013). Apparently, the algorithms running the music discovery engine are identical, independent of whether one uses the Free or the Premium service. The only difference is that advertisements play in the Free version which also cannot stream higher audio qualities.

Interestingly, most comments online regarding the ways that music discovery and recommendations at Spotify works refer to a presentation that former Tech Lead, Erik Bernhardsson, gave almost exactly at the time The Echo Nest was purchased. In his talk, Bernhardsson addressed the ways in which Spotify discovery engine functions: "How do we structure music understanding? How do you teach music to machines?" Essentially, Bernhardsson listed five ways to achieve apt recommendations: "Editorial tagging, Audio analysis, Metadata, Natural language processing [and] Collaborative filtering" (Bernhardsson 2014). The last idea determines listeners' preferences from historical use data, since constant feedback to Spotify is implicit in all streaming behaviour. When someone on Quora asked what data points Spotify Radio uses, Bernhardsson answered that they mainly deployed collaborative filtering through "large scale data mining of user logs . . . to create a statistical model of what artists/albums/track are similar . . . [which is] then post-processed and exposed in the radio service" (Bernhardsson 2013).

It is hard to tell what role collaborative filtering has in relation to Spotify Radio. Most likely different forms of user data analyses are combined. Then again, collaborative filtering is interesting since it is content agnostic. The computational strategy is to look only at user consumption patterns, so the same type of collaborative filtering models can be used to recommend books, films, or music. Since reliance is put on usage data only, popular content will be easier to recommend (compared to ignored content) simply because there is more user data. Collaborative filtering is one way to tap the collective intelligence of Spotify's millions of users, turning their preferred music and taste into a data layer to personalise everyone's experience. Then again, as Sander Dieleman has noted, collaborative filtering algorithms specific to music need to pay attention to heterogeneity of content "with similar usage patterns". As mentioned, it is a difficult method to deploy. Users may, for example, listen to entire albums in "one go, but albums may contain intro tracks, outro tracks, interludes, cover songs and remixes. These items are atypical for the artist in question, so they aren't good recommendations" (Dieleman 2014, Dieleman et al. 2013).

Reverse Engineering Radio Loops – Intervention Set Up

To answer the research questions postulated in the introduction to this article, we and system developers Roger Mähler and Johan von Boer set up an experiment with the purpose to examine several Spotify Radio loops. The loops were constructed using 160 so-called bot listeners. A bot is a small software application that runs automated tasks (or scripts), and we implemented our bots in the Python programming language. Bots appear to be human (at least to the Spotify Web client), which is why they are interesting to use as research informants (Snickars & Mähler 2017).

Our bots were Spotify Free users with literally no track record. They had "heard" no music before they were put into action. We were thus not primarily interested in the personalised recommendations Spotify's algorithms offered, but rather how Spotify Radio functioned generically. The reason was also practical, as providing our bots with a personal track record would have been hard (if not impossible) to accomplish. In addition, as virtual informants, our bots did not explicitly collect information. They were programmed and designed to search for a track, retrieve subsequent songs, partially interact, and importantly log all data caused by different actions.

As one of our aims with the intervention was to study repetitiveness in loop patterns, another hypothesis was that the size and structure of radio loops might depend on music genre as well as popularity. Hence, we decided to let our bots "listen" to both a hit song and a less popular track (albeit with some contextual similarity) as our bots would start a radio channel based on Swedish music from the 1970s. The two 24-hour interventions at Humlab took place in July and September 2016. The bot setup involved a few similar steps. First, bots were named (Anna.01. Cobolt.01, Fred.04.Mercury.04, Jane.12.Lead.01, etc.). Second, the bots –in the form of virtual users acting as research informants within the Spotify Web client – were programmed to "listen" to all subsequent songs that the Spotify Radio algorithm(s) generated. That is, we executed a major and a minor intervention. In the first round, the bots (120 in all, although many failed because of different technological problems) started Spotify Radio based on the highly popular Abba song "Dancing Queen" (released in 1976, with some 65 million streams at Spotify). In the second round, the bots (40 in all, with three failures) used the significantly less popular Swedish progressive rock band Råg i Ryggen's "Queen of Darkness" (released in 1975, with approximately 10,000 streams) to start a radio channel.

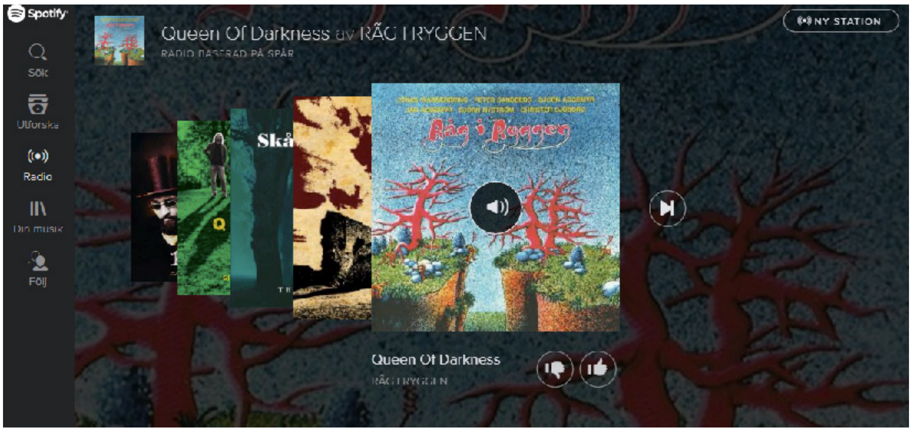

Illustration 1. The Web interface of Spotify Radio with subsequent tracks based on progressive rock band, Råg i Ryggen's "Queen of Darkness", including the three characteristic ways of adjusting recommendations: thumbs up, thumbs down, and skip.

All bots documented all subsequent tracks played in the radio loop, and importantly interacted differently within the Spotify Web client as an "obedient" bot listener, a "liker", a "dis-liker", and a "skipper". These interactions were documented, including tracks and artist played as well as breaks for advertisement. The user scenario (given to the programmers) for the first round of bots generically read as follows:

> **User scenario 1:** (approximately 30 obedient listener bots): Starts a radio station based on Abba's "Dancing Queen". Bots passively listen to the full loop. Run time 24 hours. If the radio loop stops playing, bots should be prepared to restart.
>
> **User scenario 2:** (approximately 30 liker listener bots): Starts a radio station based on Abba's "Dancing Queen". Bots like every fifth song. Run time 24 hours. If the radio loop stops playing, bots should be prepared to restart.
>
> **User scenario 3:** (approximately 30 disliker listener bots): Starts a radio station based on Abba's "Dancing Queen". Bots dislikes every fifth song. Run time 24 hours. If the radio loop stops playing, the bots should be prepared to restart.
>
> **User scenario 4:** (approximately 30 skipper listener bots): Starts a radio station based on Abba's "Dancing Queen". Bots skip every fifth song. Run time 24 hours. If the radio loop stops playing, the bots should be prepared to restart.

To introduce randomness among our bot listening behaviours, the second round of bots (the bots playing Råg i Ryggen's "Queen of Darkness") would be done in a more haphazard way, not following the strict metric logic of the first scenario. Tentative results from the first bot intervention indicated that the regular scheme was unnecessarily rigorous—i.e., too non-human. We desired more varied results, so rather than regularly liking, disliking, or skipping every fifth song, the second round of bots were programed with a 50 percent probability that they would like, dislike, or skip every third song.

Interactive feedback generated by dislikes, likes, and skips is important to Spotify. Furthermore, likes and dislikes are default user interactions on many contemporary platforms such as Facebook, YouTube, or Twitter. However, in a music streaming environment, the skip is arguably of most significance: "The skip button is now a big part of the overall listening experience" Paul Lamere, the Director of Developer Platform for The Echo Nest, stated in May 2014. Lamere's blog post was centred around how people use "the skip button when listening to music". As an insider, he has access to a vast pool of listener data. For his study, Lamere processed several billions of plays from millions of unique listeners. Basically, his data suggested that when users are engaged with music, they tend to skip more, but when music is in the background, "such as when we are working or relaxing, we skip less . . . The big surprise for me is how often we skip. . . . nearly every other song that we play" (Lamere 2014).

Our bot interventions cannot really be compared to Lamere's massive analyses; his scale is different, and as media researchers we had to work outside the data filled black boxes of Spotify and The Echo Nest. Then again, our experiment bears some resemblance with their focus on user activity and music listening behaviours that might alter recommendations. Regarding the functionality of Spotify Radio, one comment on Lamere's blog post (from reader "Bill") was particularly striking:

> As a radio programmer [your claim] backs up what we've known and how we've programmed radio stations for years. Because we are BROADCASTERS trying to be MASS APPEAL, this explains why songs are repeated so often (a familiar and popular song has the best chance of keeping most of the listeners) and placement of promos & commercials and when and how often DJ's talk and what they talk about. Our rating service . . . shows us exactly when people tune out/in and for how long in real time. That's an eye opener. Our studies also show that people will say they want new music and new music discovery but we watch data that says listeners choose familiarity almost 100% of the time. New, unfamiliar music has a very high skip rate (Bill 2014).

One of the conclusions from our interventions is in fact that artist and to some extent song repetition within Spotify Radio is reminiscent of rotation policy at commercial radio stations. The synergies between record labels and commercial radio naturally have a long and intricate media history. Suffice it to say, constant reiteration and the repeated airing of a limited playlist of songs is a central part of this relation. Songs put on heavy rotation can sometimes be played more than ten times a day so listeners are never far away from the biggest hit at the moment. Still, whether track repetition on Spotify Radio has strict commercial reasons (like at commercial radio broadcasters) remains concealed. Again, as stated, Bas Leijder Havenstroom's claim—"I believe record labels pay Spotify to have their artists to show up in radio stations"—cannot be objectively answered. The argument if song recommendation within Spotify Radio is commercial and familiarity biased is simply impossible to prove since Spotify does not share data and statistics on the matter. However, our data suggest it might be true.

Experiment Results

The two Humlab interventions used 160 bot listeners. All bot experiments ran for 24 hours. Chart 1 below displays the amount of plays of 40 Jean Lead bots that all listened to a radio channel based on Abba's "Dancing Queen". Importantly, these 40 bots were divided into ten smaller groups each with the different characteristics of an obedient listener, a liker, a dislike, and a skipper. The setup was similar for the Fred Mercury bots, the Jane Lead bots, and the Carrie Aluminium bots (in the second intervention with a radio channel based on Råg i Ryggen's "Queen of Darkness"). Taken together, our two bot interventions played a substantial amount of songs on Spotify. In the case of the 40 Jean Lead bots, they listened to more than 7,000 tracks (not including almost 1,700 ads). With some variations, the same goes for the other bot rounds. In the second intervention, with a radio channel based on "Queen of Darkness", the 40 Carrie Aluminium bots played more than 4,600 tracks. In both of our experiments, 22,624 songs were played. In addition, our bots were programmed to log all 8,367 advertisements featured within Spotify Free.

The substantial amount of plays performed by our 160 bots gives some indications as to how Spotify Radio's music discovery engine works. Working with bots has its benefits, but it is also problematic. In a former experiment linked to our project we had many problems deploying bots as informants (Snickars & Mähler 2017). In the new experiments with Spotify Radio, our systems were way more robust. They had been tested by frequent and repeated runtimes (and increased knowledge around client logic). Regarding the reliability of our interventions one

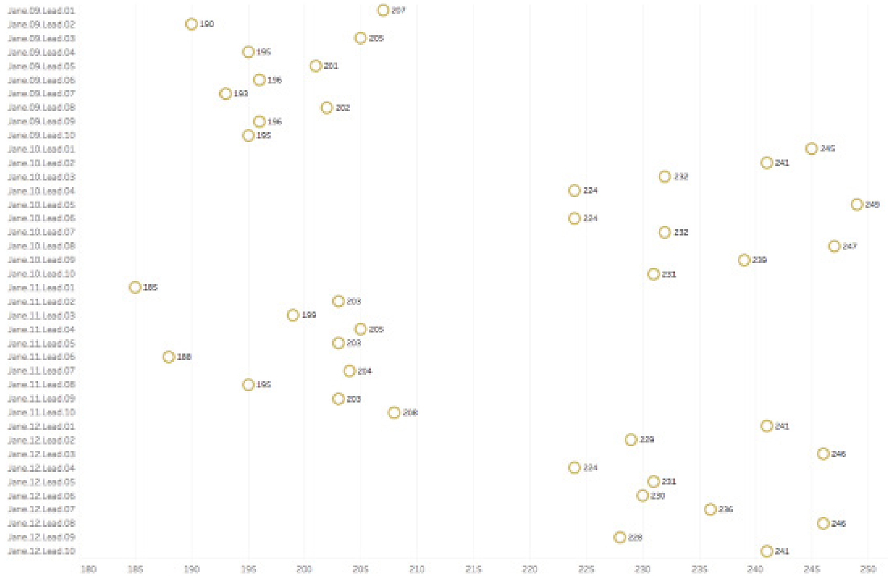

Chart 1. Total amounts of songs played by different Jane Lead bots that started a radio channel based on Abba's "Dancing Queen", ranging from 185 to 249 tracks.

should be hesitant, but what can be claimed in the statistics displayed in various charts below indeed supports assertions regarding how Spotify Radio functions. In general, song loops within Spotify Radio tend to be repetitive. Empirically our experiments demonstrate that tracks, especially the same artist, are frequently repeated. As is evident from Chart 1, average sizes of radio loops (during a 24-hour intervention) also vary substantially. Of the 40 Jane Lead bots, each bot listened to approximately 217 tracks (i.e., the average size of these radio loops). Some radio loops played by the Jane Lead bots, however, consisted of around 200 subsequent tracks and others of more than 240 tracks. Basically, the number of songs depended on the length of tracks as well as number (and length) of advertisements.

It is important to stress that due to different execution failures, which were computationally identified, all of our bots did not function properly. In both of our interventions, there were 12 failed bot rounds (nine in the first and three in the second). For some unknown reason, this was particularly the case with the 40 Fred Mercury bots, with nine failed play outs – still, the flamboyant lead vocalist of Queen did at least set the name standard for our subsequent metallic bots. A more severe problem was caused by a software bug, which made the 40 Anna Cobolt bots slightly malfunction, not performing exactly what they had been programmed to do. They basically functioned as they should, but they did not log precisely all interactions in a consistent manner. On the one hand, bugs in the Python script made bots with the dislike characteristic fail to log the very song that was disliked and instead logged the next track. On the other hand, Anna Cobolt bots with the like and dislike characteristic also failed to log the song after an advertisement had appeared. As a result, we realised that statistics from the Anna Cobolt bots deviated slightly from other bot rounds. The Anna Cobolt bots with the other two characteristics (obedient and skipper bots), however, functioned as they should. Hence, in some cases, as is evident in the charts below, results from these specific Anna Cobolt bots have been taken into statistical and comparable accounts.

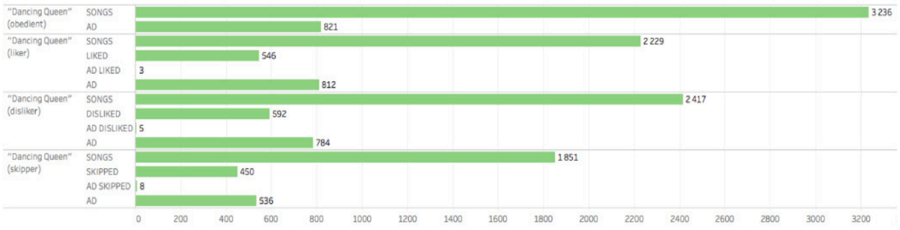

Chart 2. Number of songs, advertisements, and interactivity (dislikes, likes, and skips) among approximately 80 bots listening to a radio station based on Abba's "Dancing Queen".

Nevertheless, because of difficulties in measuring results, we decided to compare statistics within the two bot rounds by excluding the Anna Cobolt bot round as well as the other failed sessions with "invalid bot listening" and to ignore them in our overall statistical analysis. Therefore, in the first round, playing a radio station based on Abba's "Dancing Queen", our 111 bots prompted more than 13,000 different forms of content to be played, including songs and advertisements. As is evident from Chart 2, the amount of music tracks varied substantially between the different bot characteristics. The obedient bot listeners played more than 3,200 tracks, while the skipper bots only played 1,850 songs. The amount of plays between the liker and disliker bots were more equal. The same goes for the number of advertisements (around 800, but considerable less for the skipper bots, with only 500). Interestingly, the obedient bot listeners counted about the same number of advertisements even though these bots played many more songs. One general observation from the first intervention is that user interaction (like, dislike, and skip) within Spotify Free seems to have triggered more advertisements at least relative to passive bot listeners who did no interaction at all.

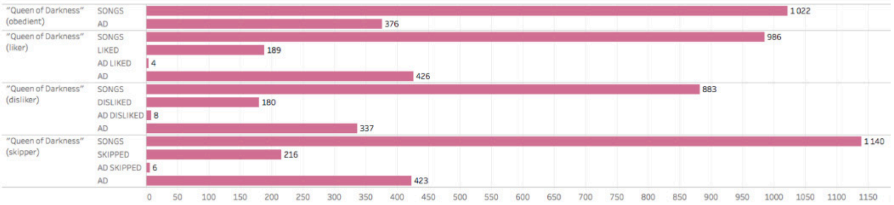

Chart 3. Number of songs, advertisements, and interactivity (skips, dislikes, and likes) among approximately 40 bots listening to a radio station based on Råg i Ryggen's "Queen of Darkness".

The second round of bots, which started Spotify Radio based on Råg i Ryggen's "Queen of Darkness", displayed a similar pattern. Again, the intervention ran for 24 hours, involving 37 bots, and almost 5,600 different forms of content were played, including songs and advertisements. As is evident in Chart 3, the number of advertisements, however, was generally higher. Between 35 to 40 percent of content played were advertisements (compared to the first bot round with around 30 percent advertisements). In relation to the first intervention, it is also striking that the skipper bots, not the obedient bots, were the ones with most songs played. This difference also suggests the difficulty in making general claims about the ways in which the Spotify Radio algorithms work. Notably, however, the second intervention played considerably fewer tracks (approximately 170 songs), compared to the first bot round (approximately 217 songs). The reason for this is mainly that progressive rock (i.e., the similar genre tracks preferred by Spotify's radio algorithms) seems to favour extended tracks, so songs were longer and fewer of them were played during a day (and night).

Another way to compare the statistics is to look at how often a song was repeated in the same playlist, i.e., to study repetitiveness in loop patterns. The six charts below depict the amount of repetitions of the same track – first, Abba's "Dancing Queen" and second, Råg i Ryggen's "Queen of Darkness" – within radio loops based on the same song. The charts depict the names of our bots with the four different characteristics in recurrent colours: the obedient (red), the liker (blue), the disliker (yellow), and the skipper (green). Again, as stated since the Anna Cobolt bots with the liker and disliker characteristics malfunctioned, these bot characteristics cannot be compared and have been excluded.

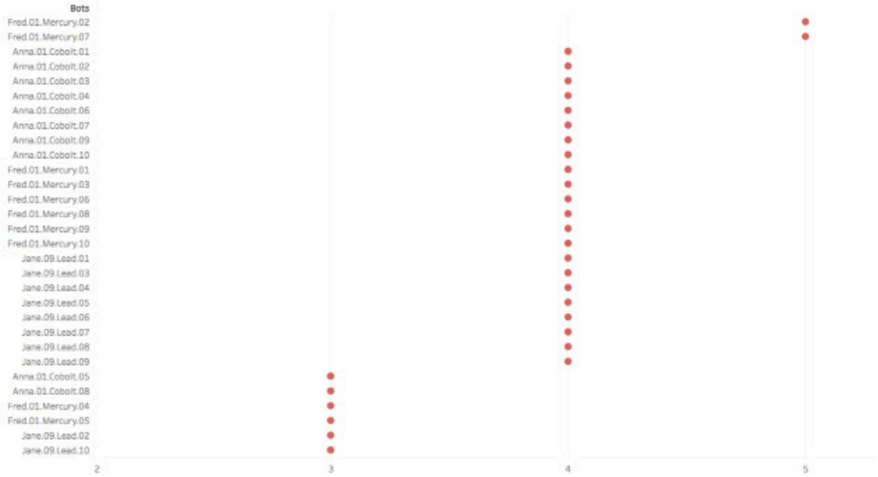

Chart 4. Amount of repetitions of Abba's "Dancing Queen" within playlists of obedient bot listeners that started a radio channel based on the same track.

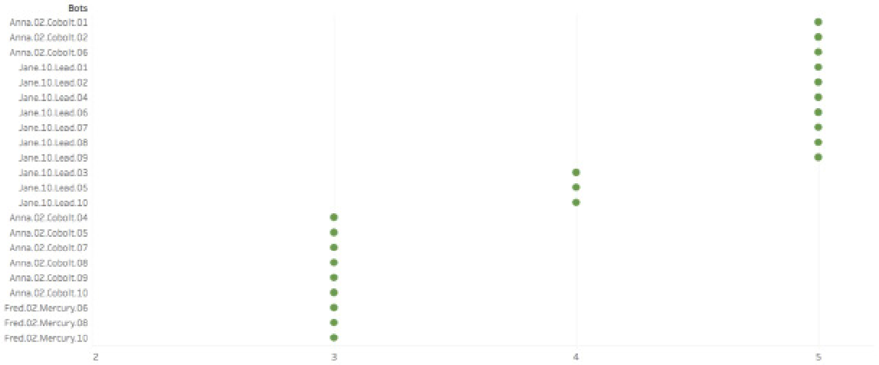

Chart 5. Amount of repetitions of Abba's "Dancing Queen" within playlists of skipper bot listeners that started a radio channel based on the same track.

In general, the songs “Dancing Queen” and “Queen of Darkness” repeatedly keep returning among the lists of tracks played by the (unknown) algorithm(s) running Spotify Radio, from twice to five times, and interestingly almost entirely independent of bot characteristics. “Dancing Queen” is repeated more often than “Queen of Darkness”, usually around four times in each radio loop. Statistics for each bot playing “Dancing Queen” reveal that the song kept returning after approximately 50 to 60 tracks. “Queen of Darkness” was repeated less often. Still, on average it kept being repeated around three times in each bot playlist.

As is apparent from charts 4–9, “Dancing Queen” was repeated more often than “Queen of Darkness” within our experiments. Another difference between the bot rounds were the general group of artists that Spotify’s radio algorithm(s) generated. In the case of Abba, nearly all recommended artists were strikingly similar, belonging to a homogenous genre of popular hit music from the 1980s (and the late 1970s). The radio stations based on “Queen of Darkness”, however,

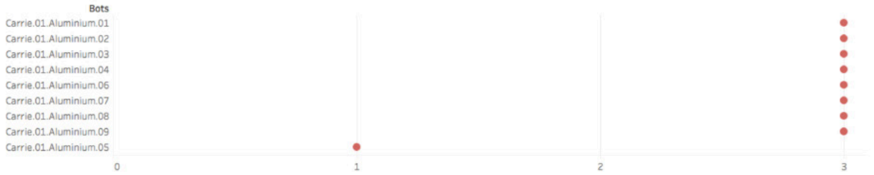

Chart 6. Amount of repetitions of Råg i Ryggen’s “Queen of Darkness” within playlists of obedient bot listeners that started a radio channel based on the same track.

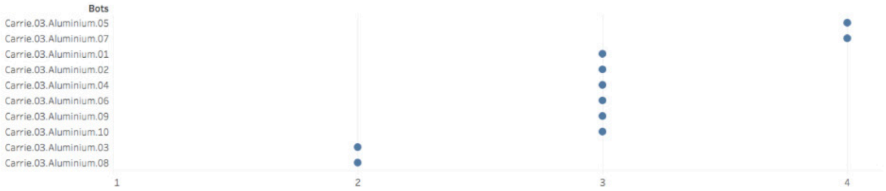

Chart 7. Amount of repetitions of Råg i Ryggen’s “Queen of Darkness” within playlists of liker bot listeners that started a radio channel based on the same track.

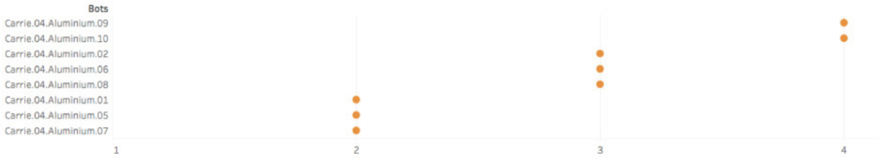

Chart 8. Amount of repetitions of Råg i Ryggen’s “Queen of Darkness” within playlists of disliker bot listeners that started a radio channel based on the same track.

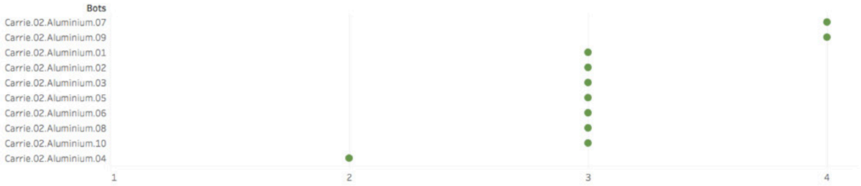

Chart 9. Amount of repetitions of Råg i Ryggen’s “Queen of Darkness” within playlists of skipper bot listeners that started a radio channel based on the same track.

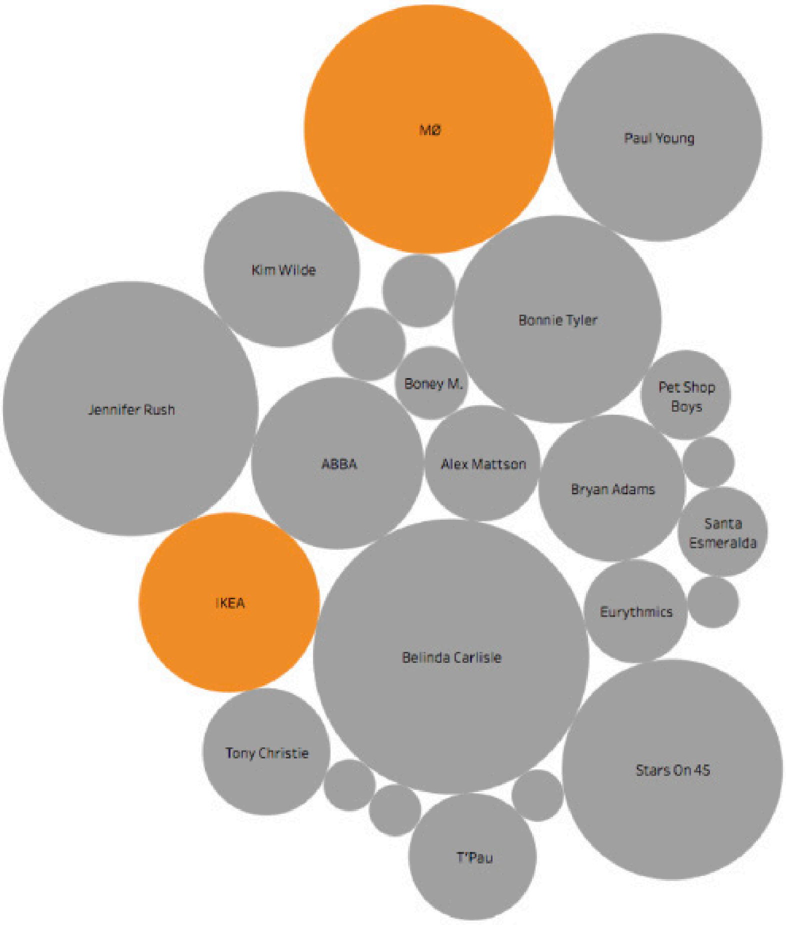

Chart 10. Artists and advertising (orange) in the Abba radio station playlist of liker bot Jane.11.Lead.02.

displayed a much greater variety in terms of artists and songs, and importantly so also from other periods than the 1970s. Songs from the rock band Mamont, for example, were released after 2010, albeit with references to the sounds of 1970s heavy classic rock, progressive, blues, and psychedelic music. The same goes for

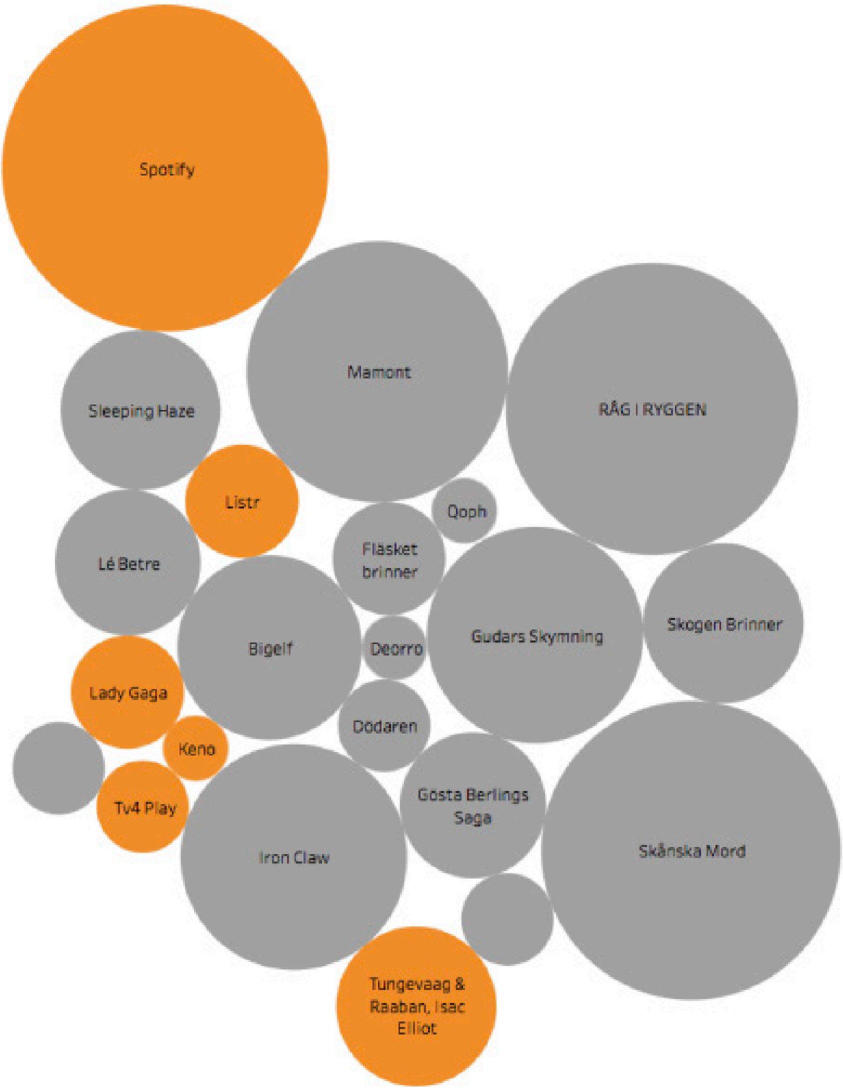

Chart 11. Artists and advertising (orange) in the Råg i Ryggen radio station playlist of liker bot Carrie.03.Aluminium.05.

Bigelf, a progressive rock and metal band formed in Los Angeles in 1991, as well as for the contemporary Swedish rock bands Gudars skymning and Skånska Mord (formed in 2006), who both play music influenced by 1970s hard and heavy rock. Temporally, radio stations based on Abba were clearly situated in the late 1970s and early 1980s, while radio stations based on Råg i Ryggen featured a greater variety of music from different periods.

A third dissimilarity between the two bot rounds was the number of advertisements. The two charts above (10 and 11) generically depict advertisement patterns in two radio loops played by two liker bots executed by bots with the most frequent repetitions of "Dancing Queen" and "Queen of Darkness". As is evident from the size of the circles, advertisement forms a substantial part of content (dis)played within Spotify Free. During the first bot's playlist, commercials include brands such as Ikea, but also promotions of artists such as Danish singer-songwriter Mø's "Final Song". As is evident in Chart 11, more and diversified advertisements were prompted in the second bot playlist, based on (the less popular) "Queen of Darkness". The playlist featured advertising for brands like Listr, TV4 Play, and Keno, as well as promotions for artists such as Lady Gaga and Tungevaag & Raaban. A general comparison between the two radio loops also makes it evident that Spotify self-promotes its Premium service repeatedly in the playlists of "Queen of Darkness", with up to 25 ads during the entire intervention, but this heavy promotion was not the case for the Abba playlist.

Chart 12 provides a final way to visualise the quite substantial amount of repetitions apparent within our experiments with Spotify Radio. Chart 12 depicts different song loops that the liker bot Jane.11.Lead.02 listened to after starting a radio station based on "Dancing Queen"; 203 tracks and 42 advertisements were played. The Spotify Radio algorithm prompted "Dancing Queen", displayed in the middle of the chart although hard to read, to be repeated five times. That is, "Dancing Queen" was played repeatedly as song number 1, 60, 77, 128, 190, and 195, resulting in two minor loops (between song number 60-77 and between 190-195) as well as three major loops. Songs by Abba were played 11 times in the loop, but the most frequent artist constantly reappearing were commercially successful and popular artists from the 1980s such as Belinda Carlisle (28 times), Jennifer Rush (24 times), Paul Young (16 times), Bonnie Tyler (16 times), and Stars On 45 (16 times, plus two promotions).

Even if the song "Dancing Queen" was repeated five times in the loops above, general data from our two bot interventions suggest that loop patterns were not that repetitive, at least not when it came to specific songs. If one takes a closer look at our data around track repetition and how often a song was repeated in the same playlist in the three complete bot playlists, few tracks were played repeatedly. Among the Fred Mercury bots, only six tracks were played five times, and

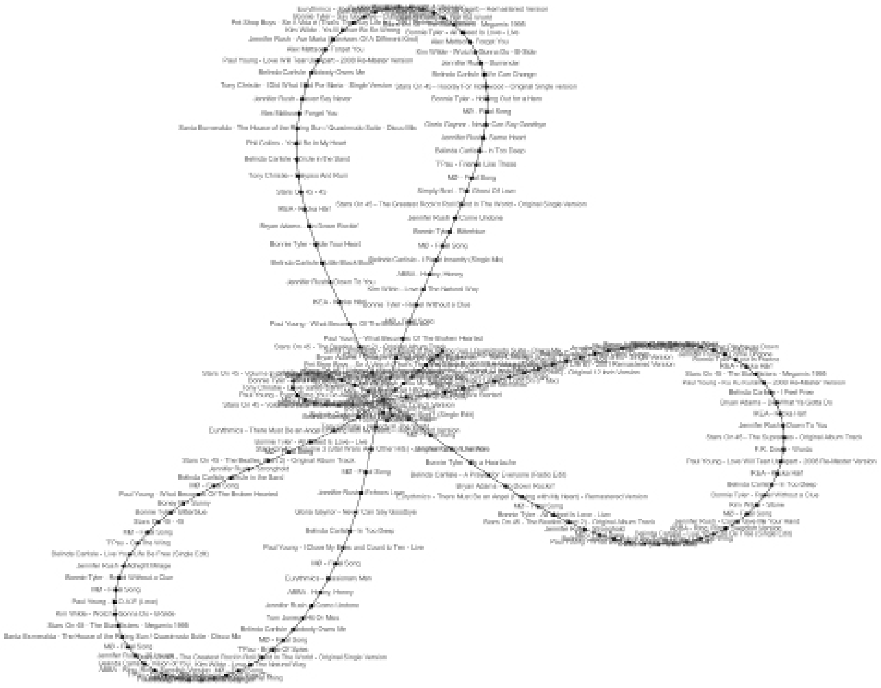

Chart 12. Different song loops as listened to by liker bot Jane.11.Lead.02. The radio channel started with Abba's "Dancing Queen" (in the middle) with the track being repeated an additional five times (during a 24-hour intervention).

1,990 tracks played one time. For the Jane Lead bots, only two tracks were played six times, but 2,628 tracks were played one time. Within the second intervention, using "Queen of Darkness" to start a radio channel, even less repetitions occurred. Among the Carrie Aluminium bots, 12 tracks were played four times (but no song was repeated more than that). In addition, 2,263 songs were played once.

Repetitions of artists within the algorithms running Spotify Radio is another matter. A closer look at the data behind Chart 12 reveals that the Spotify Radio algorithm picked 20 artists during a 24-hour playout, but only 13 artists were repeated more than twice. Given that Spotify has millions of artists to choose from, it is a relatively low figure. Even without statistically comparing the playlists of all the other bots, artistic patterns within our first bot round was also strikingly similar. Belinda Carlisle, for example, was repeated 32 times by liker bot Jane.11.Lead.07, 28 times by disliker bot Fred.04.Mercury.03, 24 times by skipper bot Fred.02.Mercury.08, and 30 times by obedient bot listener Jane.09.Lead.03. And even if the data for the malfunctioning 39 Anna Cobolt bots cannot exactly be statistically compared, their pattern looks similar as Belinda Carlisle was repeated by all of them.

Given that Spotify has millions of artists to choose from, it appears, at least at first glance, that its radio algorithm(s) did a poor job of playing a variety of artists. Hence, annoyed users at Quora and the Spotify Web Community were partly right: "The terrible radio algorithm repeats the same songs over and over (see [the linked] thread, which has been going for 2+ years)" (tellure 2015) and "need to update the algorithms for Radio, the repetitions are SAD at this point within 1 hour I can easily hear the same song three times" (zaliad 2016). Some songs were indeed recurrent, but it was foremost artist repetition that characterized Spotify Radio in our experiments. Then again, if one bears in mind the ways that Spotify Radio is reminiscent of rotation policies at commercial radio station, repetitions of popular songs and artist can also be perceived as maintaining the status quo. Reiteration is then simply default since regular listeners tend to like similar artists. As stated earlier in this article, such assumptions reveal a normative claim that to work properly the Spotify Radio algorithm should produce apt music recommendations. Perhaps it is sufficient to state that the algorithm(s) behind Spotify Radio are one-sided, since they rarely promote novel artists. Yet, on the other hand, they can also be perceived as commercially biased, although not in a pejorative way but rather from the perspective of mass appeal, as a way of keeping as many listeners as possible tuned to a station.

Conclusions

One general result from our Humlab bot interventions is that, as part of understanding the inner workings of a central contemporary music delivery platform, it is indeed possible to measure loop patterns on Spotify Radio. This study's results suggest that bits can be used as a set of concrete methodologies for performing humanist inquiry on big data and black-boxed media services as Spotify, media that today increasingly serve as key delivery mechanisms for cultural materials. The bot logs also made it possible to empirically sustain claims of repetitiveness within Spotify Radio and indeed prove that at least artist iterance is quite striking. The regularity of patterns is in fact prominent, and music loops are definitively not endless. On the contrary, they display a repeated pattern with slight variations depending on which artist a radio station was based on as well as (to lesser extent) bot characteristics. The tracks that we based our radio stations on, for example, kept returning in the bot playlists. If a radio loop started with "Dancing Queen", after some 50 tracks, it was played again by the Spotify Radio algorithm(s). Bots listening to a radio station based on "Queen of Darkness" displayed a similar tendency, albeit with the difference that the song was not repeated as often as "Dancing Queen" and at shorter intervals (regularly after some 70 tracks or so).

One conclusion to draw from our experiments with Spotify Radio is that similar artists reappear frequently within all bot playlists. It seems that music recommendation algorithms do not take advantage of the archival infinity at Spotify. In short, fans of Belinda Carlisle will be pleased, since a radio station based on "Dancing Queen" will repeatedly play her songs no matter what kinds of interactions are executed. Essentially, the same goes for radio stations based on "Queen of Darkness", where few artist (yet from different periods) constantly appeared as well.

If Spotify Radio is about personalisation of content, as the company claims, then the recommendation algorithms are a disappointment. This type of algorithmic critique, which the present article can (at least to some extent) empirically verify, taps into contemporary discussions around the ways that machines are altering our taste (for the worse). "Spotify is making you boring", music journalist Scott Timberg argued in an issue of *Salon* during the summer of 2016. "With all the songs at our fingertips, we're exposed to very few, thanks to how digital recommendations work", he stated, which sounds like a description of our interventions. Algorithms, Timberg further wrote, influence listener habits "whether driving your streaming playlists, your Amazon recommendations, or suggestions on iTunes – are about driving you closer and closer to what you already know." Instead of taking users "toward what you want to listen to, they direct you toward slight variations of what you're already consuming. (2016)

Timberg's journalistic speculations, based on personal experiences of massive amounts of music listening, essentially describes most of the results from our bot interventions, at least on a more general level. Radio stations based on our sample songs did result in slight variations of similar music content. One apparent difference, however, between the two bot rounds was that the "Queen of Darkness" radio stations featured a greater variety of music in terms of different periods even if fewer similar artists kept being played. Another difference was that more and diversified advertisements were featured in the Råg i Ryggen bot playlists. In our data, however, we could not really detect specific differences between radio stations based on "Dancing Queen" and "Queen of Darkness". Our hypothesis that the size and structure of radio loops might depend on music genre as well as popularity was not supported. Indeed, we should have investigated more music genres and more frequently tracked repetition, but artist reiterations indeed were frequent. During both bot interventions, Spotify Radio constantly kept playing more of the same.

An even more troubling result, at least for Spotify, is that radio loops tend to look the same, independent of bot characteristics. Adjusting Spotify Radio through user recommendations such as thumbs up (like), thumbs down (dislike), or skip did, in short, *not* produce differences in results. Although Spotify Radio boasts that "[t]he more you personalise the stations to match your tastes the better they get", this is hardly the case, given data from our experiments. At the Spotify Community Web similar comments are also frequent, actually with some users proving their complaints with tangible data. For example, user hahndreas claims that "[g]iving a thumbs-down for songs doesn't prevent them to be played again just a few songs after. . . . If in a radio-station I'm constantly skipping and thumbing-down slow songs, why doesn't it move to faster stuff? This should be basic behaviour, shouldn't it?" (2015). User tamar makes her/his complaint even more directly: "WTF, Spotify?" (2015).

In general, song loops with bots programmed as obedient listeners contained more tracks regarding both "Dancing Queen" and "Queen of Darkness" in our experiments. Otherwise, the data suggest very few real differences between the various bot characteristics. Therefore, an important result from our experiments is that music loop patterns basically look the same, even if bot interaction were performed differently within the Spotify Web client as an obedient bot listener, a liker, a disliker, and a skipper. One reason for this might be that our bots did not have any track record. Following such logic, without previous plays, Spotify's recommendation engine would have difficulties fine-tuning suggestions and music tastes. However, if one remembers Bas Leijder Havenstroom remark that even if he started a radio station "based on a playlist with many, many artists", his experience was still "that some (specific) artists keep coming back". Fine tuning the radio functionality was beyond the bounds of possibility, and the same goes for our interventions. A radio channel based on "Dancing Queen" (nearly) always generated the following artists, regardless of bot characteristics: Bryan Adams, Belinda Carlisle, Toni Christie, Jennifer Rush, Bonnie Tyler, and Paul Young. "Like and unlike buttons are purely decorative", was user xebec-us' characterisation of Spotify Radio. xebec continued: "Nothing really gets liked or unliked – you'll hear the same exact songs with the same frequency as if you were just skipping them" (2015). Even our introduction of randomness among the bot listening behaviours in radio stations based on "Queen of Darkness" did not alter the repeated results generated by the algorithm; more of the same music kept returning.

Another more general conclusion from our experiments is, hence, that the recommendation ability of Spotify Radio is exaggerated. In fact, one might even argue that the claim of musical personalisation and the ability to be recommended an infinity of content to some extent is even untrue. "The more you personalise the better the music gets" should rather be perceived as a mendacious company advertisement used to attract listeners and create commercial interest in the radio functionality (at least when the service was novel). Since complaints were made right after the launch of Spotify Radio, it is likely that the recommendation functionality was flawed from the start. In fact, former Tech Lead, Erik Bernhardsson, said in a lecture in 2013 that "learning from feedback is actually pretty hard" (Bernhardsson 2013). In addition, the Spotify intern and researcher, Sander Dieleman, also stated that the user feedback that Spotify collects through "thumbs up and thumbs down that users can give to tracks played on radio stations" is hard to measure and make use of and "[t]his type of information is very useful to determine which tracks are similar. Unfortunately, it is also quite noisy" (Dieleman 2014).

Finally, this article concludes that the various forms of public critiques of the inadequate functionality of Spotify Radio are (and were) spot on. In short, it is a service that has not functioned particularly well at least not as a music recommendation system. By and large, however, Spotify seems to have been aware of its malfunctioning radio service and continued to neglect the issue (for different reasons). In an article in *Wired* from May 2016 centred around the latest music discovery releases (Discover Weekly and Release Radar), Edward Newett, the company's lead software engineer and the person who coded Discover Weekly, problematized the issue: "At the time, I don't think we were super focused on music discovery in that sense. . . . Spotify had the Discover page, and the artist and song radio, and that seemed good enough" (2014). The interview (together with other accounts) gives the impression that when Spotify Radio was released, it was a premature technology that tried to give listeners good recommendations, but ultimately failed.

Today, the technology seems to have caught up although not in a radio setting. In fact, the radio metaphor within music recommendation systems has been increasingly modified, and recently it has even been altered in favour of other analogies: "We now have more technology than ever before to ensure that if you're the smallest, strangest musician in the world, doing something that only 20 people in the world will dig, we can now find those 20 people and connect the dots between the artist and listeners", Matthew Ogle, who oversees Discovery Weekly at Spotify, stated in an interview (Pasick 2015). The main ingredient in Discover Weekly is collaborative filtering of user playlists where human selections and groupings of songs form the core of service recommendations. During the last year, Spotify has put way more emphasis on Discover Weekly and recently Release Radar than on its radio functionality. In short, there seems to be a specific tech-musical recommendation narrative, stretching from Spotify Radio (2011) to Discover Weekly (2014) to Release Radar (2016). Unlike Discover Weekly, Release radio's tracks are brand new and have no listening data. Instead, Spotify relies on a solution that tries to predict who will enjoy a song by analysing the audio signal.

In the end, it seems that traditional radio recommendations appear to be less significant for Spotify, at least in comparison to new types of machinic suggestions. Naturally, feedback data from listener activity and user profiles remain essential for Spotify's music recommendation systems. Yet, given that the medium of radio has been around since the 1920s, it is perhaps not surprising that the old radio metaphor (and its cloud counterpart around the celestial jukebox) has finally been superseded. For many years, listeners saw streaming services as a way of getting to know new music; however, as streaming music and not radio became the default listening mode, it is hardly unexpected that the radio metaphor would lose its popularity and consequently be replaced with new computational recommendation formats based on taste profiles, song identification, and digital fingerprints.

Pelle Snickars is Professor of Media and Communication Studies, specialising in digital humanities at Umeå university, and is affiliated with the Humlab research centre. His research focuses on the relationship between old and new media, media economy, digitisation of cultural heritage, media history as well as the importance of new technical infrastructures for the humanities.

References

Allen Anderson, Paul (2015): "Neo-Muzak and the Business of Mood", *Critical Inquiry* 41:4, 811-840.

Bernhardsson, Erik (2013): "What data points does Spotify's Radio feature use?" https://www.quora.com/What-datapoints-does-Spotifys-Radio-feature-use (Accessed 15/03/17).

Bernhardsson, Erik (2014): "Music Discovery at Spotify" http://www.slideshare.net/erikbern/music-recommendations-mlconf-2014 (Accessed 15/03/17).

Bill (2014) https://musicmachinery.com/2014/05/02/the-skip/ (Accessed 15/03/17).

Chandra, Vikram (2013): *Geek Sublime: Writing Fiction, Coding Software,* London: Faber.

Dieleman, Sander (2014): "Recommending music on Spotify with deep learning" http://benanne.github.io/2014/08/05/spotify-cnns.html (Accessed 15/03/17).

Dieleman, Sander, van den Oord, Aaron & Schrauwen, Benjamin (2013): "Deep content-based music recommendation" https://biblio.ugent.be/publication/4324554/file/4324567 (Accessed 15/03/17).

Fixmer, Andy (2012): "Spotify Said Developing Pandora-Like Online Radio Service" http://www.bloomberg.com/news/articles/2012-04-26/spotify-said-developing-pandora-like-online-radio-service (Accessed 15/03/17).

Gehl, Robert W. (2014): *Reverse Engineering Social Media. Software, Culture, and Political Economy in New Media Capitalism,* Philadelphia: Temple University Press.

Gunnarsson Lorentzen, David (2016): *Following Tweets Around Informetric methodology for the Twittersphere,* Borås: The Swedish School of Library and Information Science.

Hachman, Mark (2012): "Spotify to Take On Pandora With Radio Service" http://uk.pcmag.com/internet-products/64346/news/report-spotify-to-take-on-pandora-with-radio-service (Accessed 15/03/17).

hahndreas (2015): "The radio option sucks" https://community.spotify.com/t5/Newcomers-and-Contribution/The-radio-option-sucks/td-p/863585/page/4 (Accessed 15/03/17).

Friesinger, Günther & Herwig, Jana (2014) (eds.): *The Art of Reverse Engineering,* Bielefeld: Transcript.

Hietala, Heikki (2014): “Why do my Spotify radio sounds so repetitive?” https://www.quora.com/Why-do-my-Spotify-radio-sounds-so-repetitive (Accessed 15/03/17).

Hu, Yajie. & Ogihara, Mitsunori (2011): “NextOne Player: A Music Recommendation System Based on User Behavior”, *12th International Society for Music Information Retrieval Conference*, 103–108.

Kitchin, Rob (2016): “Thinking critically about and researching algorithms”, *Information, Communication & Society*, 24 February, 1-16.

Kjus, Yngvar (2016): “Musical exploration via streaming services: The Norwegian experience”, *Popular Communication,* 14: 3, 127-136.

Lamere, Paul (2014): “The Skip” https://musicmachinery.com/2014/05/02/the-skip/ (Accessed 15/03/17).

lehwark (2012): “Better Radio Algorithms” https://community.spotify.com/t5/Closed-Ideas/Better-Radio-Algorithms/idi-p/107485 (Accessed 15/03/17).

Leijder Havenstroom, Bas (2015): “Why does my Spotify Radio play the same artists over and over for me?” https://www.quora.com/Why-does-my-Spotify-Radio-play-the-same-artists-over-and-over-for-me (Accessed 15/03/17).

Lin, Ning, Tsai, Ping-Chia, Chen, Yu-An & Chen, Homer H. (2014): “Music Recommendation Based on Artist Novelty and Similarity”, *Multimedia Signal Processing, IEEE 16th International Workshop,* Jakarta, 1-6.

Madrigal, Alexis C. (2014): “How Netflix Reverse Engineered Hollywood” *The Atlantic* January 2 https://www.theatlantic.com/technology/archive/2014/01/how-netflix-reverse-engineered-hollywood/282679/ (Accessed 15/03/17).

Modell, Amanda (2015): “Mapping the Music Genome: Imaginative Geography in Pandora Internet Radio and the Genographic Project”, *Media Fields Journal,* 10, 1-13.

Morris, Jeremy Wade & Powers, Devon (2015): “Control, curation and musical experience in streaming music services” *Creative Industries Journal* 2, 106-122.

Pasick, Adam (2015): “The magic that makes Spotify’s Discover Weekly playlists so damn good” *Quartz* December 21 http://qz.com/571007/the-magic-that-makes-spotifys-discover-weekly-playlists-so-damn-good/ (Accessed 15/03/17).

Popper, Ben (2015): “How Spotify’s Discover Weekly cracked human curation at internet scale” http://www.theverge.com/2015/9/30/9416579/spotify-discover-weekly-online-music-curation-interview (Accessed 15/03/17).

Quora (2016): “Spotify Radio” https://www.quora.com/search?q=Spotify+radio (Accessed 15/03/17).

Resare, Noa (2013): “How does Spotify radio work” https://www.quora.com/How-does-Spotify-Radio-work (Accessed 15/03/17).

Seaver, Nick (2014a): “On Reverse Engineering. Looking for the cultural work of engineers” https://medium.com/anthropology-and-algorithms/on-reverse-engineering-d9f5bae87812#.9tw36ldh1 (Accessed 15/03/17).

Seaver, Nick (2014b): “Knowing algorithms” https://static1.squarespace.com/static/55eb004ee4b0518639d59d9b/t/55ece1bfe4b030b2e8302e1e/1441587647177/seaverMiT8.pdf (Accessed 15/03/17).

Seaver, Nick (2016) http://nickseaver.net/ (Accessed 15/03/17).

Shao Dingding, Bo & Tao Li, Wang (2009): “Music Recommendation Based on Acoustic Features and User Access Patterns” *IEEE Transactions on Audio, Speech, and Language Processing* 17:8,1602–1611.

Snickars & Mähler, Roger (2017): “SpotiBot—Turing Testing Spotify” *Digital Humanities Quarterly* forthcoming.

Spotify Radio (2016) https://support.spotify.com/us/using_spotify/search_play/spotify-radio/ (Accessed 15/03/17).

Tamar (2015): “The radio option sucks” https://community.spotify.com/t5/Newcomers-and-Contribution/The-radio-option-sucks/td-p/863585/page/4(Accessed 15/03/17).

tellure (2015): "'Only play songs Spotify has never played before' option for Radio" https://community.spotify.com/t5/Closed-Ideas/All-Platforms-Radio-quot-Only-play-songs-Spotify-has-never/idi-p/1225168 (Accessed 15/03/17).

The Echo Nest (2011): "The Echo Nest Powers Spotify Radio" http://blog.echonest.com/post/14311681173/spotify-radio-the-echo-nest (Accessed 15/03/17).

The Echo Nest (2014): "Spotify Acquires Echo Nest" http://the.echonest.com/press-releases/spotify-acquires-echo-nest/ (Accessed 15/03/17).

Timberg, Scott (2016): "Spotify is making you boring: When algorithms shape music taste, human curiosity loses" *Salon* June 10 http://www.salon.com/2016/06/10/spotify_is_making_you_boring_when_algorithms_shape_music_taste_human_curiosity_loses/ (Accessed 15/03/17).

Whitman, Brian (2012): "How music recommendation works—and doesn't work" http://notes.variogr.am/post/37675885491/how-music-recommendation-works-and-doesnt-work (Accessed 15/03/17).

xebec-us (2015): "The radio option sucks" https://community.spotify.com/t5/Newcomers-and-Contribution/The-radio-option-sucks/td-p/863585/page/4 (Accessed 15/03/17).

zaliad (2016): "Better Radio Algorithms lead to better music" https://community.spotify.com/t5/Closed-Ideas/All-Platforms-Radio-Better-Radio-Algorithms-lead-to-better-music/idi-p/1333406 (Accessed 15/03/17).