Music Recommendation and the Long Tail

Mark Levy
Last.fm
Karen House
1-11 Baches Street, London N1 6DL, UK
mark@last.fm

Klaas Bosteels
Last.fm
Karen House
1-11 Baches Street, London N1 6DL, UK
klaas@last.fm

ABSTRACT

Using a dataset of 7 billion recent submissions to the Last.fm Scrobble API[1], we investigate possible popularity bias in Last.fm's recommendations and streaming radio services. In particular we compare the recent listening of users who listen regularly to Last.fm streaming services with those who listen less often or never. Finally we describe a new service explicitly designed to make recommendations from the long tail, and analyse popularity effects across the recommendations which it suggests.

1. INTRODUCTION

Music lovers today have access to a previously undreamed of quantity and variety of recordings. Music is available through an increasing number of digital channels, including free online streaming services, "all you can eat" subscription services, and paid downloads, not to mention via illegal downloading and more traditional physical media. In one well publicised view [2], this proliferation in availability should lead to a reduction in the dominance of hits in our musical culture. With the development of advanced tools for search and recommendation, we should expect to see listeners discovering and enjoying a huge range of music that may be less popular overall, sitting somewhere in the so-called *Long Tail* of sales ranks, but which offers a good match for their own personal tastes.

The original long tail speculation was that it would become increasingly profitable to "sell less of more" by making large numbers of niche items easily available. Empirical studies of consumer behaviour suggest that this is indeed true, provided that enough choice is available, and that effective search and recommendation systems are provided to help users find their way around large inventories [4, 3]. A large recent study of consumer preference data, including user ratings for movies and music, shows that while not all users consume items in the long tail, "the vast majority of users are a little bit eccentric, consuming niche products at least some of the time", in particular reporting high average ratings for niche music [9].

Two lines of research suggest, however, that the utopian vision in which niche movies and music increasingly usurp the dominance of hits may not be borne out in practice. A recent study of the Netflix catalogue of movies shows that, on the contrary, demand for hits appears to rise, while that for niche products falls, as the number of available titles increases [12]. Meanwhile hit products continue to dominate the consumption of movies and music even for users who regularly explore the long tail [6]. Secondly, a number of studies of the very recommender systems which are supposed to support discovery in the long tail suggest that such systems are frequently prone to *popularity bias*, recommending globally popular items ahead of niche products [5, 7, 1].

In this paper we present an empirical study of the recommendations actually made by the widely-used Last.fm music recommender system, in particular via its streaming radio service, and set them in the context of wider music listening. As well as assessing the degree of popularity bias in these recommendations, we also compare the listening habits of a large group of music lovers regularly exposed to Last.fm's streaming radio with those of a second group who have no exposure to it. Finally we outline the design of a recommender system expressly designed to make recommendations from the long tail, and assess the popularity bias of a sample of the recommendations it produces.

The remainder of this paper is organised as follows: Section 2 briefly reviews previous work on popularity bias in recommender systems; Section 3 describes the data used as the basis for this study; Section 4 investigate the presence of popularity bias in Last.fm's radio streams, and Section 5 attempts to uncover any corresponding influence on users' wider listening habits; Section 6 outlines a music recommender explicitly designed to make recommendations in the long tail, and Section 7 draws conclusions.

2. PREVIOUS WORK

Three recent studies identify potential bias in recommender systems, particularly those based on *collaborative filtering* (CF). In [1] recommendations are generated using various well-established CF algorithms based on movie ratings from the MovieLens and Netflix datasets[2]. Over 84% of the MovieLens recommendations were for movies in the top 20% by

[1] http://www.last.fm/api/submissions

[2] http://www.grouplens.org, http://www.netflixprize.com

WOMRAD 2010 Workshop on Music Recommendation and Discovery, colocated with ACM RecSys 2010 (Barcelona, SPAIN)

number of ratings. No comparable figure is given for the Netflix recommendations, but the authors suggest that in both cases large gains in the diversity of recommendations can be achieved, with little cost to relevance, by suitable reranking techniques applied to the CF output.

The effect of recommendations on user behaviour is studied in a completely simulated setting in [7]. In the simulation users receive and consume recommendations from a CF recommender over a series of timesteps, so that over time the recommendations they receive are influenced by the previously recommended items which were added to their profile in previous rounds. The simulation was run repeatedly, with differing outcomes, but led in the great majority of runs to a decrease in the overall diversity of consumption.

CF recommendations are studied indirectly in [5], which considers the network defined by Last.fm's similar artist relationships. These relationships provide one of the sources of data used in Last.fm's recommender system, and can also be directly navigated as links on the Last.fm website, providing an active form of music discovery. Besides observing that Last.fm's similar artist lists are dominated by other artists with a similar level of popularity, [5] computes various network metrics to support the assertion that "CF tends to reinforce popular artists, at the expense of discarding less-known music", essentially by showing that navigation from popular to long tail artists often involves traversing a large number of artist links.

While all three of these papers discuss the effects of CF recommender systems, none of them considers a dataset of real recommendations made by a deployed system. In this paper we use Last.fm submissions data, defined fully in the next Section, to study the effect of a large-scale recommender system in practice.

3. DATA

Last.fm allows music lovers to *scrobble* details of their music listening. Scrobbling is available from media players and streaming services either though native support or via a suitable plugin, and is built in to some hardware devices. The Scrobble API[3] supports the submission of various events: in this paper we distinguish between *radio listens*, which record the act of playing a track via one of Last.fm's own streaming radio stations, and *scrobbles*, where the track played comes from any source other than Last.fm. In both cases the submitted metadata includes an artist name: for a scrobble this is typically drawn from the ID3 tags of the track being played.

Last.fm provides various types of streams, including *Similar Artists* and *Tag* radio, launched by supplying a seed artist or tag respectively, available to anyone, and *Recommendation* and *Library* radio, available to any user registered for scrobbling. Recommendation radio plays tracks by artists selected for the user by Last.fm's recommender, while Library radio plays tracks by artists previously scrobbled by the user. Users typically listen to Last.fm's radio stations through the flash player on the Last.fm website, or via a client program on their computer, phone or games console. In each case, information is displayed about the artist of the current track, including links to the artist's page on the Last.fm website, lists of similar artists, etc. While Recommendation radio is clearly an explicit recommendation service, all the stations can be considered as offering implicit recommendations, with Similar Artists radio in particular relying on underlying similarity data which also forms part of the input to Last.fm's recommender system. Even Library radio can be regarded as providing a form of non-novel recommendation, as it may remind the user of artists whom they like but have not listened to for some time.

[3]http://www.last.fm/api/submissions

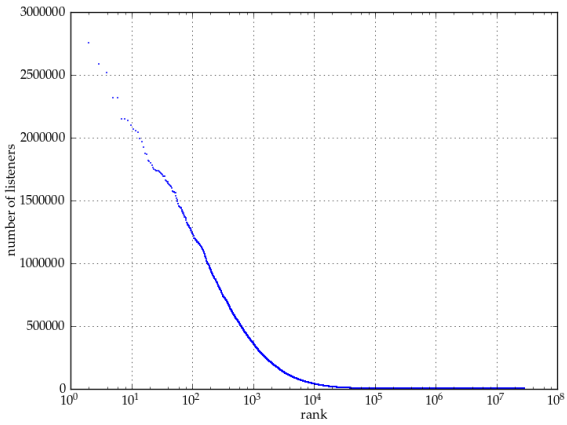

Figure 1: Artist popularity amongst Last.fm users.

In the following analysis we therefore pay special attention to Recommendation radio, but also consider the influence of Last.fm streaming radio as a whole. For the time being we neglect the influence of the recommendations displayed on users' Last.fm home pages and dedicated recommendation pages. The dataset used consists of over 7 billion submissions to the Scrobble API received between January and May 2010.

4. POPULARITY BIAS

The most widely-used measure of the diversity or, conversely, *concentration* of a set of products consumed by a group of users is the Gini coefficient [8], and this has also been applied to measure popularity bias within recommendations [7]. The Gini coefficient is computed from the area bounded by the Lorenz curve, which, in the case of artist recommendations, plots the proportion of the total number of recommendations made cumulatively for the bottom x% of artists recommended. The Gini coefficient is not ideal for our purposes here, as it depends on artist ranks within the set of recommendations being evaluated, i.e. it would show high concentration for a recommender that overwhelmingly recommended a small number of artists, even if all the artists it recommended belonged to the long tail. In the Sections that follow we therefore show plots similar to Lorenz curves, but showing the cumulative proportion of recommendations made in relation to artist ranks according to their *global* popularity, based simply on the overall total number of scrobbles received at the time of writing, shown in Fig. 1. We can also use this data to define what we mean by a "long tail" artist. Fitting Kilkki's informal model [10] suggests that this is any artist below rank 20,000; Fig. 1 shows, however, that in reality popularity flatlines slightly further down the tail, and that a reasonable definition of a long tail artist is one at rank 50,000 or below.

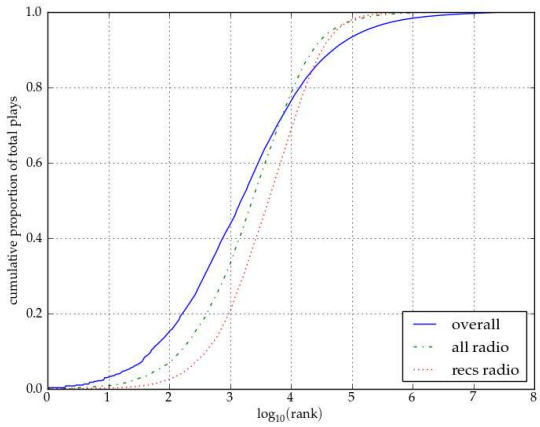

Figure 2: Popularity bias in Last.fm radio. Cumulative plays by artist rank for Recommendation radio and for all Last.fm radio stations. For comparison we also show the cumulative proportion of all scrobbles received during the same period.

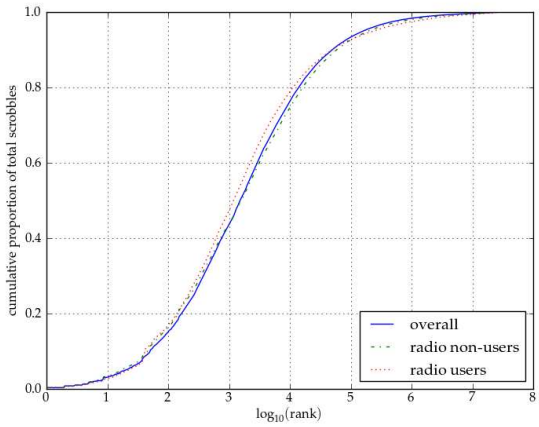

Figure 3: Possible influence of Last.fm radio. The plots show the cumulative proportion of scrobbles received by artist rank for two groups of users, one regularly exposed to Last.fm radio and the other completely unexposed to it.

Fig. 2 shows the distribution of ranks for artists played on Last.fm Recommendation radio, and on all Last.fm radio stations, compared with the distribution for all tracks scrobbled in same period. We observe that Last.fm radio is somewhat biased away from hit artists in comparison to the listening of Last.fm users as a whole, while Last.fm's recommendations are even more strongly biased towards lower ranking artists. In particular we see that artists in the top-1000 of overall listening make up 40% of scrobbles but only 20% of plays on Recommendation radio. Recommendation radio plays the same proportion of long tail artists as are listened to overall, but includes fewer plays of the lowest ranked artists: it is reasonable to assume, however, that artists scrobbled at those ranks include many whose tracks are not readily available for streaming, as well as spurious artists based on submissions with incorrect metadata that is not repaired by Last.fm's automatic correction system.

5. INFLUENCE

To expose the possible influence of Last.fm recommendations on users, we first create a set of active listeners by taking all users who registered during the five months under consideration and then scrobbled at least 500 but no more than 20,000 tracks during that time. The upper limit removes spammers and other technically-minded enthusiasts whose scrobbles represent a superhuman quantity of listening within that period, while the lower limit ensures that we have a reasonable amount of listening data for all of the users under consideration. We then draw two samples from this set of listeners. The first contains all users who had no exposure to any Last.fm radio station within the period (or indeed at any stage, as we include only newly-registered users). The second group contains all users for whom radio listens made up 25-75% of their submissions, i.e. these listeners are highly exposed to Last.fm radio, but also make a significant number of scrobbles for listening outside Last.fm.

Fig. 3 shows the distribution of artists scrobbled by each of these groups in the first five months of 2010, again compared with that for all tracks scrobbled during the same period. We observe a bias towards more popular artists in the mid region for the group of radio listeners, but it is small compared with the biases in artist popularity for radio plays shown in Fig. 1, and, more importantly, clearly not correlated with them. To control for demographic or other systematic differences between users who listen frequently to radio and those who never do so, in Fig. 4 we compare scrobbles for users with low exposure to radio, making up 10-50% of their scrobbles, to those with radio making up 50-90% of their scrobbles. In contrast to Fig. 3, this shows a slight bias towards the long tail in users with higher exposure to radio. We can conclude that there is no evidence that radio and recommendations cause a systematic bias towards more popular artists.

6. LONG TAIL RECOMMENDATIONS

We build a prototype recommender for long tail artists using conventional item-based CF. We first identify a suitable candidate pool of long tail artists from which to draw our recommendations. For each artist in our overall catalogue we then find the most similar k artists within the pool, based on scores computed by comparing both scrobbles and tags applied to each artist. When a user u requests recommendations, we create a profile of artist weights W_u based on their scrobbles, and build up a candidate set containing the top-k similar artists in the pool for each artist in W_u. We then score each candidate artist a based on their similarity to artists in the user's profile, computing a score $P_{u,a}$ using the well-known weighted sum method [11], finally returning the top-N highest scoring artists:

$$P_{u,a} = \sum_{a' \in W_u} sim(a, a') w_u(a') \quad (1)$$

where $sim(a, a')$ is the similarity between a and a', and $w_u(a')$ is the weight assigned to a' in the the user profile.

To obtain a suitable pool of long tail artists, we start with all artists with tracks currently available in the Last.fm "Play direct from artist" scheme, under which unsigned artists or labels holding suitable rights can make tracks available for

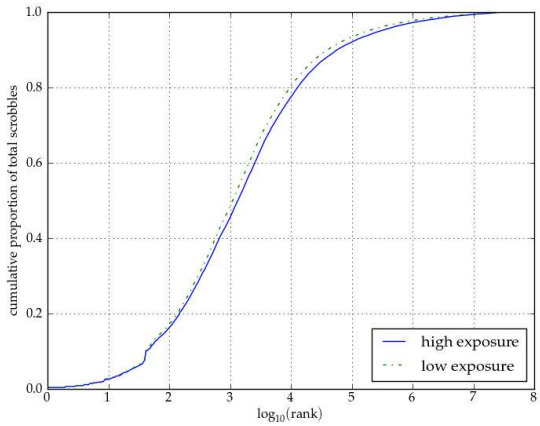

Figure 4: Scrobbles received for users with low and high exposure to Last.fm radio respectively.

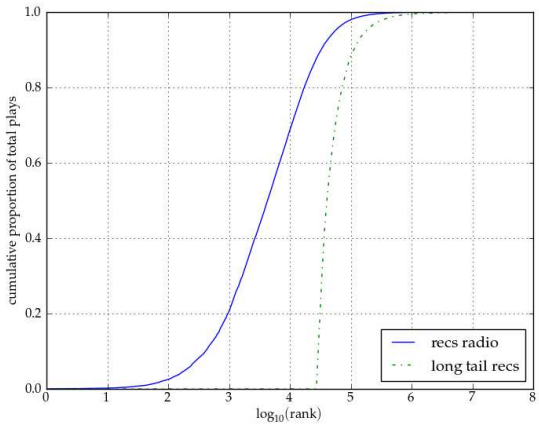

Figure 5: Long tail recommendations vs plays on the main Recommendation radio.

free streaming from their Last.fm pages. This scheme is aimed at niche and new artists, but to be sure that artists in the pool are indeed from the long tail, we also apply a hard cutoff on current overall reach, removing any artists with more than 10,000 listeners. Finally to mitigate problems with artist disambiguation in the long tail, where new or niche artists have the same names as more popular artists, we mine Last.fm wiki entries for key phrases indicating multiple artists with the same name, removing any affected artists from the pool. The resulting set of long tail artists is updated daily, but at the time of writing contains 118,000 artists.

To study the popularity distribution amongst artists suggested by this new system, we generate 50 recommendations for each of a sample of 100,000 active Last.fm users, defined as users who have visited the Last.fm website within the last week. Fig. 5 shows the resulting distribution, compared to that for plays on the main Recommendation radio station during the first five months of 2010. Approximately 90% of the sampled recommendations are for artists in the mid to long tail, with ranks 25,000 to 100,000, with the remaining 10% being for the lowest ranking artists. While the previous Section suggests that the influence of recommendations may be limited, we can reasonably hope that the prototype recommender will gradually stimulate increased interest in the long tail.

7. CONCLUSIONS

A comparative analysis of artists chosen by Last.fm's recommender system and a large body of listening data suggests that, contrary to claims in the literature based on laboratory experiments, real world music recommenders do not necessarily exhibit strong popularity bias. Our results suggest that, in any event, the influence of such a recommender on users' general listening may be limited. Finally we sketch the design of a prototype recommender designed explicitly to suggest artists from the long tail. Future work includes a user evaluation of the prototype system, which is now publicly available[4].

[4]http://playground.last.fm/demo/directrecs

8. REFERENCES

[1] G. Adomavicius and Y. Kwon. Toward more diverse recommendations: Item re-ranking methods for recommender systems. In *Proc. 19th Workshop on Information Technologies and Systems*, 2009.

[2] C. Anderson. *The Long Tail. Why the future of business is selling less of more.* Hyperion, 2006.

[3] E. Brynjolfsson, Y. Hu, and D. Simester. Goodbye pareto principle, hello long tail: The effect of search costs on the concentration of product sales. Technical report, MIT Center for Digital Business, 2007.

[4] E. Brynjolfsson, Y. Hu, and M. Smith. From niches to riches: anatomy of the long tail. *Sloan Management Review*, 47(4):67–71, 2006.

[5] O. Celma and P. Cano. From hits to niches? or how popular artists can bias music recommendation and discovery. In *Proc. 2nd Netflix-KDD Workshop*, 2008.

[6] A. Elberse. A taste for obscurity? an individual-level examination of "Long Tail" consumption. Technical report, Harvard Business School, 2007.

[7] D. Fleder and K. Hosanagar. Recommender systems and their impact on sales diversity. In *EC '07: Proceedings of the 8th ACM conference on Electronic commerce*, 2007.

[8] C. Gini. Measurement of inequality of incomes. *The Economic Journal*, 21(121):124–6, 1921.

[9] S. Goel, A. Broder, E. Gabrilovich, and B. Pang. Anatomy of the long tail: ordinary people with extraordinary tastes. In *WSDM*, 2010.

[10] K. Kilkki. A practical model for analyzing long tails. *First Monday*, 12(5), 2007.

[11] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, 2001.

[12] T. Tan and S. Netessine. Is Tom Cruise threatened? using Netflix Prize data to examine the Long Tail of electronic commerce. Technical report, Wharton Business School, University of Pennsylvania, 2009.