WHAT'S HOT ? ESTIMATING COUNTRY-SPECIFIC ARTIST POPULARITY

Markus Schedl[1], Tim Pohle[1], Noam Koenigstein[2], Peter Knees[1]

[1] Department of Computational Perception
Johannes Kepler University, Linz, Austria

[2] Faculty of Engineering
Tel Aviv University, Tel Aviv, Israel

ABSTRACT

Predicting artists that are popular in certain regions of the world is a well desired task, especially for the music industry. Also the cosmopolitan and cultural-aware music aficionado is likely be interested in which music is currently "hot" in other parts of the world. We therefore propose four approaches to determine *artist popularity rankings* on the country-level. To this end, we mine the following data sources: *page counts from Web search engines*, *user posts on Twitter*, *shared folders on the Gnutella file sharing network*, and *playcount data from last.fm*. We propose methods to derive artist rankings based on these four sources and perform cross-comparison of the resulting rankings via overlap scores. We further elaborate on the advantages and disadvantages of all approaches as they yield interestingly diverse results.

1. INTRODUCTION

To determine popular artists for a certain country or cultural region of the world, one can obviously look into publicly available music charts, such as the "Billboard Hot 100", released weekly for the United States of America by the *Billboard Magazine* [6]. However, this straightforward strategy is hardly feasibly when we aim at broaden the scope to the whole world. The reasons are manifold.

First, not all countries do release music charts for various reasons. Causes may be, for example, a lack of capability to determine music sales or an underdevelopment of music distribution at large. Even if data is available, it is often not publicly accessible, and even if so, not always in an easy-to-use format, e.g., via a Web service.
Second, even if charts are available for a specific country, they often cover only certain ways of music distribution. Commonly they are strongly biased towards sales figures of music albums. In some countries, however, they also include digital music sales via online stores. This inhomogeneity between countries, i.e., the in- or exclusion of certain distribution channels, make such data hardly comparable between different countries of the world. Another aspect to be considered here are possible heavy distortions caused by (illegal) music sharing channels, since legislation in this area varies severely between countries. In fact, the majority of today's music distribution is affected via file sharing networks [2]. Thus, traditional charts, such as the "Billboard Hot 100", are becoming less and less relevant.
Third, if the aim is to come up with a list of the most popular artists ever, countries lacking solid historical records constitute an obvious problem.

Summarizing these challenges, we conclude that analyzing which kind of music is popular in a specific country or cultural region necessitates taking a deeper look into various distribution channels and data sources. In this paper, we therefore present four different approaches to estimate artist popularity rankings on the country-level, each of which makes use of a different data source. The first one is based on *page count estimates* of Web search engines, the second approach analyzes *Twitter posts*, the third one derives information from meta-data of *users' shared folders in a Peer-to-Peer network*, and the fourth one uses *playcount data from last.fm*.

In the remainder of this paper we review related literature (Section 2), present four approaches to determine artist popularity on the country-level (Section 3), elaborate on the conducted evaluation experiments and discuss their results (Section 4), and finally draw conclusions (Section 5).

2. RELATED WORK

Related work falls into two categories: literature that particularly tackles the task of chart prediction, and work that relates to the four approaches we propose for this task.

Targeting the problem of predicting music charts, Koenigstein and Shavitt [26] present an approach to predict the charts based on search queries issued within the Peer-to-Peer (P2P) file sharing network *Gnutella* [35]. The authors show that a song's popularity in the P2P network highly correlates with its ranking in the Billboard charts. The authors' approach can further predict upcoming charts with high accuracy. However, for their analysis Koenigstein and Shavitt only consider the United States.
Pachet and Roy [33] try to predict the popularity of a song based on audio features and a variety of manual labels. The authors' conclusion is, however, that even state-of-the-art machine learning techniques fail to learn factors that determine a song's popularity, irrespective of whether they are trained on signal-based features or on high-level human annotations.

In [38] Schedl et al. propose several heuristics to determine which artists are popular within a certain genre. They relate occurrence counts of artist names on Web pages via an approach similar to *Google*'s backlink and forward link analysis [34]. The authors show that downranking factors for artist names equaling common speech terms improve accuracy when comparing the resulting rankings against a ground truth popularity categorization extracted from *allmusic.com* [3].
In [22] Grace et al. derive popularity rankings from user comments in the social network *MySpace* [32]. To this end, the authors apply various annotators to crawled *MySpace* artist pages in order to spot, for example, names of artists, albums, and tracks, sentiments, and spam. Subsequently, a data hypercube (OLAP cube) is used to represent structured and unstructured data, and to project the data to a popularity dimension. A user study showed that the list generated by this procedure was on average preferred to the Billboard charts.

Previous work that relates to the four approaches proposed here comprise the following.
Our heuristic that uses *page counts* returned by search engines builds upon work from [20, 39], where Web co-occurrences of artist names and terms specific to the music domain are used to categorize artists, a process also known as "autotagging" [13]. In [37] Schedl et al. propose a similar approach to estimate artist similarity. The authors suggest a simple probabilistic model that defines similarity between two artists a and b as the conditional probability of a to be mentioned on a Web page known to relate to b and vice versa. Accuracies of up to 85% were reported for genre classification.
To the best of our knowledge, *Twitter* [41] has not been scientifically investigated for music information extraction and retrieval yet. Although there do exist certain commercial services, such as *BigChampagne* [7] and *Band Metrics* [9], which seem to incorporate microblogging data into their artist and song rankings, no details on their approach are available. Furthermore, they strongly focus their services on the USA. A general study on the use of *Twitter* can be found in [24]. Java et al. report that *Twitter* is most popular in North America, Europe, and Asia (Japan), and that same language is an important factor for cross-connections ("followers" and "friends") over continents. The authors also distill certain categories of user intentions to microblog. Employing the *HITS* algorithm [25] on the network constructed by "friend"-relations, Java et al. derive user intentions from structural properties. They identified the following categories: information sharing, information seeking, and friendship-wise relationships. Analyzing the content of *Twitter* posts, the authors distill the following intentions: daily chatter, conversations, sharing information/URLs, and reporting news.
Using *Peer-to-Peer networks* as data source for music information retrieval, [8, 14, 31, 43] rely on data extracted from *OpenNap* to derive music similarity information. All of these papers seem to build upon the same data set, which comprises of metadata on shared content (approximately 3,000 shared music collections were analyzed). Logan et al. [31] compare similarities defined by artist co-occurrences in shared folders, by expert opinions from *allmusic.com*, by playlist co-occurrences from *Art of the Mix* [4], by data gathered from a Web survey, and by MFCC features [5]. To this end, they calculate a "ranking agreement score", i.e., the pairwise overlap between the N most similar artists according to each data source. The main findings are that the co-occurrence data from *OpenNap* and from *Art of the Mix* show a high degree of overlap, the experts from *allmusic.com* and the participants of the Web survey show a moderate agreement, and the signal-based MFCC measure had a rather low agreement with the music context-based data sources. More recently, in [40] Shavitt and Weinsberg mine the *Gnutella* file sharing network to derive artist and song similarities. The authors gathered metadata of shared music files from about one million *Gnutella* users in November 2007, which yielded information on half a million songs. Analyzing the 2-mode graph of users and songs revealed that most users share similar files. The authors further propose a method for artist recommendation based on the gathered data.
Taking a closer look at the data source of *music information systems*, which corresponds to the fourth approach, not only *last.fm* [28] provides popularity rankings via their API [29]. *Echonest* [15] offers a function to retrieve a ranking based on the so-called "hotttness" of an artist [17]. This ranking is based on editorial, social, and mainstream aspects [16]. However, this Web service does not provide country-specific information.

3. DETERMINING ARTIST POPULARITY ON THE COUNTRY LEVEL

We propose the following four heuristics to determine an artist's popularity in a certain country, and consequently create an artist popularity ranking. To this end, we first retrieve a list of 240 countries from *last.fm* [30], based on which the following approaches operate.

3.1 Search Engine Page Counts

This approach makes use of a search engine's number of indexed Web pages for a given query, a count usually referred to as *page count*. These page counts are, however, only rough estimates of the real number of available Web pages related to the query. Nevertheless, for the purpose of classifying music artists into genres [20, 37, 39] and for classifying general instances according to a given ontology as well as for learning sub- and superconcept relations [11, 12], this method yielded respectable results.

For the paper at hand, we queried the search engines *Google* [21] and *Exalead* [18], using their API or issuing HTTP requests. The page count values returned for all $\langle$artist, country$\rangle$ tuples were retrieved. To avoid excessive bandwidth consumption, we restrict the number of search results to be transmitted to the smallest value (this is usually one result). Since we are only interested in the page count estimates, this restriction effectively reduces network traffic without effecting the results.
The two main challenges of this approach are directing the search towards pages related to the music domain and alleviating the distortions caused by artist names that equal common speech words. We address these issues by using queries of the form

```
"artist name" "country name" music
```

and weighting the resulting page count values with a factor resembling the *inverse document frequency (idf)* [46]. The final ranking score is thus calculated according to Formula 1, where $pc_{c,a}$ is the page count value returned for the country-specific query for artist a and country c, N is the total number of countries for which data is available, and df_a is the number of countries in which artist a is known according to the data source (i.e., the number of countries with $pc_{c,a} > 0$).

$$popularity\_pc_{c,a} = pc_{c,a} \cdot \log_2 \left(1 + \frac{N}{df_a}\right) \quad (1)$$

3.2 Twitter Posts

Many *Twitter* posts reveal information about what people are doing or thinking right now. We are interested in posts containing information about which music is currently being played by users in a given country. To accomplish this, we retrieve posts using the *Twitter* Search API [42]. The posts are then narrowed in two ways. First, we only search for posts containing the hashtag *#nowplaying*. This restriction is directly supported by the *Twitter* API. As a second restriction, the search is narrowed to a specific country. Not being aware of a more direct implementation for the second restriction, we search only for posts whose users are located within a certain radius around a GPS coordinate. More specifically, for a given country, we determine the coordinates of larger cities (with more than 100,000 inhabitants) and search for posts originating from a circle of 100 kilometers around the respective coordinates. The names of the cities are taken from *Wikipedia*, e.g., [45], and the coordinates are determined by using *Freebase* [19]. For each city location for which geolocation data is resolved successfully, all *Twitter* posts available through the *Twitter* API are retrieved, which yields a maximum of about 1,500 posts per city location.

One of the advantages of using this kind of data is certainly its recentness. Thus, the retrieved data may contain artists that do not appear in our list of most popular artists (cf. Section 4.1). A first look at the format of the texts reveals that automatic tokenization seems not easily to accomplish due to the large variation of wording and creative methods to use the available number of characters. We therefore opt to scan the retrieved texts for the artists contained in the artist list, and we count the number of their appearances for a given country c. This count equals the term frequency ($tf_{c,a}$) of a in an aggregated document comprising all posts gathered for cities in country c. Formula 2 gives the ranking score. The rightward term again represents an *idf*-factor that downranks artists that are popular everywhere, and thus not specific to country c. N is the total number of countries, and df_a is the number of aggregated country documents in which artist a occur.

$$popularity\_twi_{c,a} = tf_{c,a} \cdot \log_2 \left(1 + \frac{N}{df_a}\right) \quad (2)$$

3.3 Shared Folders in a P2P Network

Collecting shared folder data from *Gnutella* users is a two-staged-process. First, a *crawler* needs to discover the current network topology (which is very dynamic). Subsequently, a *browser* queries the active users for their shared folders data. The crawler treats the network as a graph, and performs a breadth-first exploration, where newly discovered nodes are enqueued in a list of un-crawled addresses. The crawler provides a list of active IP addresses to the browser, which sends *Gnutella* “Query” messages [1] to the clients. The clients reply with “QueryHit” messages, that lists their shared folder content. These messages are the basis for our P2P data set.

The system described above is a different system than the one used by Koenigstein and Shavitt in [26], which collected *Gnutella* search queries for song ranking. One advantage of a shared folder data set over queries is the availability of ID3 tags and hash keys, which simplifies the process of associating the digital content with a musical artist. However, when singles ranking is considered (as in [26]), queries tend to better reflect the changing popularity trends of pop songs over short time intervals. In this study, we associate artists with digital content by matching the artist names against the content of the ID3 tags. Occasionally, the content in ID3 tags is missing or misspelled. We therefore, match the artists names against the file names as well.

In order to build popularity charts for specific countries, one needs to resolve the geographical location of the users. The geo-identification is based on the IP addresses. First, we generate a list of all unique IP addresses in the data set (typically over a million). We resolve the geography of IP addresses using the commercial *IP2Location* [23] database. Each IP address is bounded with its country code, city name, and latitude-longitude values. This accurate geographical information pin points artists’ fans and enables tracking spatial diffusion of artists popularity [27].

After the digital files are associated with artists names and geography, building popularity charts is straightforward. For each country, we aggregate the total number of digital content that is associated with each artist. Ranking is then performed according to frequency.

3.4 Last.fm Playcounts

We further estimate country-specific artist popularity based on the community of *last.fm* users. Despite the issues of *hacking and vandalism* as well as the *community bias* [36], which are inherent to collaborative music information systems, the playcounts of *last.fm* users can be expected to reflect to a certain extent which music is currently popular. We therefore gathered the top 400 listeners of each country at the end of 2009. We subsequently extracted the top-played artists for each of the resulting top-listeners-sets. [1] Aggregating the playcounts for each artist over a country’s top listeners finally yielded a popularity ranking for the country under consideration.

4. EVALUATION

4.1 Test Set

We used *last.fm*’s Web API [29] to gather the most popular artists for each country of the world, as of November 2009. We then aggregated this data into a single list of 201,135 unique artist names.

4.2 Experiments

As we aim at assessing the pros and cons of the various approaches, without yet having an established ground truth for this kind of experiments, we choose to perform a pairwise comparison of the approaches. Each approach produces a ranked list of artists for the various countries. Expecting that the absolute numbers obtained by the various approaches are not immediately comparable, we compare the produced artist popularity rankings of two approaches

[1] In the meantime, *last.fm* has extended its API with a `Geo.getTopArtists` function, which can be used to directly retrieve the top-played artists among a certain country’s users. Quick empirical comparisons showed that the implementation behind this function seems to resemble our approach.

A_j and A_k. This comparison is done separately for each country c. In the next subsections, we describe the applied data preprocessing steps and the used evaluation measure in detail.

4.2.1 Preprocessing

We start our analysis by processing the artist names in the artist popularity list for country c of each approach in a basic way (e.g., each artist name is represented in lower case, repeated whitespace characters are removed, and UTF-8-encoded characters are transformed to canonical ASCII representations).

Instead of using raw artist counts directly, we normalize them, attempting to avoid dominance of common-speech words, or globally popular artists whose popularity is not highly country specific. For each artist, the number of countries this artist appears in is counted. Each country-specific artist count $\mathrm{ac}_{c,a}$ is then normalized as indicated in Equation 1.

Artist names appearing in the two lists (given by the pair of approaches under investigation) are matched against each other, and only artists appearing in both lists are kept. Based on this data, we calculate the overlap between the rankings obtained with the two prediction approaches, as described next.

4.2.2 Evaluation Measures

The top-n rank overlap for country c between approaches A_j and A_k is calculated as

$$\mathrm{ro}_{c,A_j,A_k,n} = \frac{1}{n} \cdot \left|\{a|\max\left(r_{A_j,c,a}, r_{A_k,c,a}\right) \leq n\}\right| \quad (3)$$

where $r_{A_j,c,a}$ denotes the ranking of artist a in country c according to approach A_j, only considering the artists for which both approaches (A_j and A_k) yield a ranking score. In other words, the top-n rank overlap is the fraction of artists appearing within the top n ranked artists in both approaches. For example, if one artist is within the top-2 ranked artists of both approaches, the top-2 rank overlap is 0.5. Obviously, n can take values up to the number of artists $n_{\max,c}$ for which both approaches deliver rank data for country c, and the top-$n_{\max,c}$ rank overlap is always 1.

To obtain an overall measure for two approaches and a given country, we define the country-wise rank overlap as

$$\mathrm{cro}_{c,A_j,A_k} = \frac{1}{n_{\max,c}} \sum_{n=1}^{n_{\max,c}} \mathrm{ro}_{c,A_j,A_k,n} \quad (4)$$

which has a trivial (random) baseline of about 0.5 and a maximum value of 1.0 when both rankings are identical. The country-wise rank overlaps are further combined to obtain one overall scalar value for approaches A_j and A_k. To account for the different quantity of available information, we weight the overlap score of each country with the number of artists for which information is available. We define the overall overlap measure between approaches A_j and A_k as

$$\mathrm{ov}\left(A_j, A_k\right) = \frac{\sum_{c\in C} n_{\max,c} \cdot \mathrm{cro}_{c,A_j,A_k}}{\sum_{c\in C} n_{\max,c}} \quad (5)$$

The measure ov also has a trivial baseline of about 0.5 and a maximum value of 1.0.

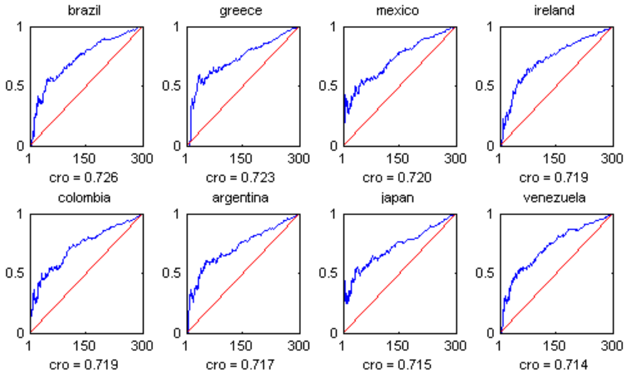

Figure 1. Top 8 countries for *pc_google* vs *p2p*.

To give an illustrative example, Figure 1 shows for the comparison of approach *pc_google* and *p2p* the 8 countries with highest ro value, as a chart from $1..n_{\max,c}$.

4.3 Results and Discussion

Each approach offers at least a slightly different view on reality since the data sources are of distinct nature. There is also no such thing as a "ground truth" for this task, as each data source (even "Billboard"-style charts) is biased, as elaborated below. Nevertheless, we would like to point out certain interesting observations.

Looking at Figure 2, the highest overlap score of 0.67 is found between *Google page counts* and *P2P*. One reason may be that the two sources have broadest coverage. Another explanation may be the time dependency. *Twitter* and *last.fm* are much more time dependent, whereas *P2P shared folders* and *amounts of Web pages* change much slower. In fact, the content of the data sources behind P2P networks and Web search engines, i.e., users' music collections and Web pages, respectively, is accumulated over years. Microblogging posts and *last.fm* data, in contrast, change much faster and are therefore more likely to reflect trends.

Second, the *page counts* approach using *Google* and the same approach using *Exalead* do not produce similar results, as we would have expected. In fact, the rankings reveal a non-significant overlap of 0.51. A possible explanation is that the two search engine providers may use very different page count estimation techniques.

Exalead shows the lowest overlap with other sources. Its highest overlap is realized, not surprisingly, with *Google* and with *P2P*, but it remains slightly above the baseline (0.53). An explanation for *Exalead*'s low overlap score becomes apparent when looking at Figure 3. *Exalead* has by far the highest number of matching artists, which may induce a high noisiness.

In terms of country coverage (cf. Figure 3), the *last.fm* and the *page counts* approaches offer data for nearly every country in the world.

To account for the different nature and scope of the proposed approaches (and underlying data sources), we compare them according to several aspects in Table 1, elaborating on specific advantages and disadvantages. One issue is that certain approaches are prone to a specific bias. For example, the average *last.fm* user does not represent the average music listener of a country, i.e., *last.fm* data is distorted by a "community bias". The same is true for *Twitter*, which is biased towards artists with very active fans. On the other hand, some very popular artists may have fans

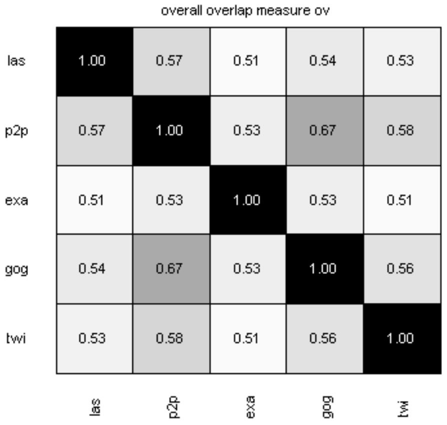

Figure 2. Overlap ov between each pair of approaches.

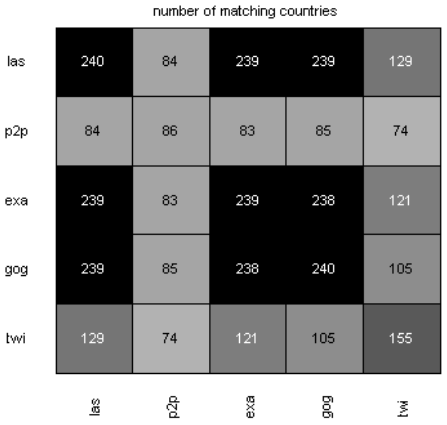

Figure 3. Number of countries with non-empty overlap.

that twitter to a much lower degree. This issue becomes especially apparent when thinking of live artists vs. dead ones: The live ones keep making new headlines, and probably also have many more active fans, while the dead ones have an inherent problem with this. Traditional charts are biased towards the data the music industry uses to derive them, usually record sales figures.

Another aspect according to which the approaches differ considerably is the availability of data. While *page count estimates* are available for all countries of the world, the *P2P* and *Twitter* approaches suffer from a very unbalanced coverage, strongly depending on the country under consideration. Also traditional music charts vary strongly between countries and continents with respect to availability. According to [44], only one country in Africa publishes official music charts, while this number amounts to 19 for Europe.

A big advantage of traditional charts is their virtual immunity against noise. *Page count estimates*, in contrast, are easily distorted by ambiguous artist or country names. *last.fm* data suffers from hacking and vandalism [10], as well as from unintentional input of wrong information and misspellings.

In the dimension of time dependence, the approaches can be categorized into "current" and "accumulating", depending on whether they reflect the instantaneous popularity, or a general, all-time popularity in that they accumulate popularity levels over time.

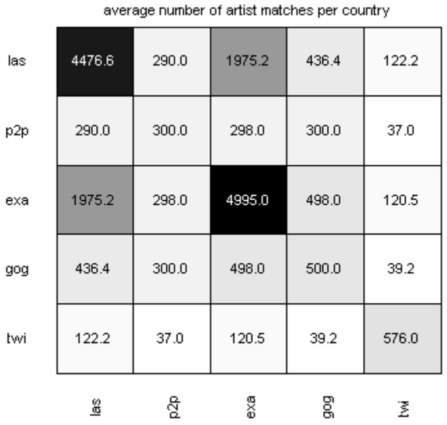

Figure 4. Average number of artists per country ($n_{\max,c}$).

5. CONCLUSIONS AND FUTURE WORK

We presented four approaches to determine country-specific artist popularity rankings based on different data sources (search engine's page counts, *Twitter* posts, shared folders in the *Gnutella* network, and playcounts of *last.fm* users). In the absence of a standardized ground truth, we performed pairwise comparison of the approaches and elaborated on particular advantages and disadvantages. Most approaches showed only weak overlaps, probably due to the different nature of their data sources. We found, however, a considerable overlap between *Google* page counts and P2P data, which is probably explained by the similar time scope the two data sources cover. As a general conclusion, we can state that artist popularity can be derived from various, quite inhomogeneous data sources. The remarkably weak overlap between most of them indicates that the quest for artist popularity is a multifaceted and challenging task, in particular in today's era of multi-channel music distribution. To derive one overall popularity measure, we will need to combine the different sources.

Future work will hence foremost aim at elaborating hybrid approaches that account for the different quantity and quality of information output by the four heuristics. We will also work on refining our approaches to capture artist popularity within certain genres, e.g., by incorporating methods similar to [38]. We will further look at the various processing steps in more detail. Most of the current implementations were created in an ad-hoc manner, and some of the choices might degrade the performance. For example, better string comparison algorithms may improve results for artists whose names may be spelled in various ways. Alternative ways of normalizing artist counts for the individual approaches are also likely to yield improvements.

6. ACKNOWLEDGMENTS

This research is supported by the *Austrian Fonds zur Förderung der Wissenschaftlichen Forschung* (FWF) under project numbers L511-N15 and Z159.

7. REFERENCES

[1] The Gnutella Protocol Specification v0.41. http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf (access: March 2010).

Source/Aspect	Bias	Availability	Noisiness	Time Dependence
Page Counts	Web users	comprehensive	high	accumulating
Twitter	community	country-dependent	medium	current
P2P	community	country-dependent	low–medium	accumulating
Last.fm	community	high	medium–high	accumulating
Traditional Charts	music industry	country-dependent	low	current

Table 1. A comparison of different approaches according to various dimensions.

[2] Digital Music Report 2009. http://www.ifpi.org/content/library/DMR2009.pdf (access: May 2010), January 2009.

[3] http://www.allmusic.com (access: January 2010).

[4] http://www.artofthemix.org (access: February 2008).

[5] Jean-Julien Aucouturier, François Pachet, and Mark Sandler. ”The Way It Sounds”: Timbre Models for Analysis and Retrieval of Music Signals. *IEEE Transactions on Multimedia*, 7(6):1028–1035, December 2005.

[6] http://en.wikipedia.org/wiki/Billboard_Hot_100 (access: May 2009).

[7] http://www.bigchampagne.com (access: May 2010).

[8] Adam Berenzweig, Beth Logan, Daniel P.W. Ellis, and Brian Whitman. A Large-Scale Evaluation of Acoustic and Subjective Music Similarity Measures. In *Proceedings of ISMIR*.

[9] www.bandmetrics.com (access: May 2010).

[10] Òscar Celma and Paul Lamere. ISMIR 2007 Tutorial: Music Recommendation.

[11] Philipp Cimiano, Siegfried Handschuh, and Steffen Staab. Towards the Self-Annotating Web. In *Proceedings of ACM WWW*, 2004.

[12] Philipp Cimiano and Steffen Staab. Learning by Googling. *ACM SIGKDD Explorations Newsletter*, 6(2):24–33, 2004.

[13] Douglas Eck, Thierry Bertin-Mahieux, and Paul Lamere. Autotagging Music Using Supervised Machine Learning. In *Proceedings of ISMIR*, 2007.

[14] Daniel P.W. Ellis, Brian Whitman, Adam Berenzweig, and Steve Lawrence. The Quest For Ground Truth in Musical Artist Similarity. In *Proceedings of ISMIR*, 2002.

[15] http://echonest.com (access: March 2010).

[16] http://developer.echonest.com/docs/method/ get_hotttnesss (access: March 2010).

[17] http://developer.echonest.com/docs/method/ get_top_hottt_artists (access: March 2010).

[18] http://www.exalead.com (access: February 2010).

[19] http://www.freebase.com (access: March 2010).

[20] Gijs Geleijnse and Jan Korst. Web-based Artist Categorization. In *Proceedings of ISMIR*, 2006.

[21] http://www.google.com (access: March 2010).

[22] Julia Grace, Daniel Gruhl, Kevin Haas, Meenakshi Nagarajan, Christine Robson, and Nachiketa Sahoo. Artist Ranking Through Analysis of On-line Community Comments. In *Proceedings of ACM WWW*, 2008.

[23] http://www.ip2location.com (access: March 2010).

[24] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why We Twitter: Understanding Microblogging Usage and Communities. In *Proceedings of WebKDD/SNA-KDD*, 2007.

[25] Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5), 1999.

[26] Noam Koenigstein and Yuval Shavitt. Song Ranking Based on Piracy in Peer-to-Peer Networks. In *Proceedings of ISMIR*, 2009.

[27] Noam Koenigstein, Yuval Shavitt, and Tomer Tankel. Spotting Out Emerging Artists Using Geo-aware Analysis of P2P Query Strings. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.

[28] http://last.fm (access: March 2010).

[29] http://last.fm/api (access: March 2010).

[30] http://www.last.fm/community/users (access: March 2010).

[31] Beth Logan, Daniel P.W. Ellis, and Adam Berenzweig. Toward Evaluation Techniques for Music Similarity. In *Proceedings of ACM SIGIR: Workshop on the Evaluation of Music Information Retrieval Systems*, 2003.

[32] http://www.myspace.com (access: November 2009).

[33] François Pachet and Pierre Roy. Hit Song Science is Not Yet a Science. In *Proceedings of ISMIR*, 2008.

[34] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bringing Order to the Web. In *Proceedings of ASIS*, 1998.

[35] Matei Ripeanu. Peer-to-Peer Architecture Case Study: Gnutella Network. In *Proceedings of IEEE Peer-to-Peer Computing*, 2001.

[36] Markus Schedl and Peter Knees. Context-based Music Similarity Estimation. In *Proceedings of LSAS*, 2009.

[37] Markus Schedl, Peter Knees, and Gerhard Widmer. A Web-Based Approach to Assessing Artist Similarity using Co-Occurrences. In *Proceedings of CBMI*, 2005.

[38] Markus Schedl, Peter Knees, and Gerhard Widmer. Investigating Web-Based Approaches to Revealing Prototypical Music Artists in Genre Taxonomies. In *Proceedings of ICDIM*, 2006.

[39] Markus Schedl, Tim Pohle, Peter Knees, and Gerhard Widmer. Assigning and Visualizing Music Genres by Web-based Co-Occurrence Analysis. In *Proceedings of ISMIR*, 2006.

[40] Yuval Shavitt and Udi Weinsberg. Songs Clustering Using Peer-to-Peer Co-occurrences. In *Proceedings of the IEEE ISM: International Workshop on Advances in Music Information Research (AdMIRe)*, San Diego, CA, USA, 2009.

[41] http://twitter.com (access: February 2010).

[42] http://apiwiki.twitter.com/Twitter-API-Documentation (access: March 2010).

[43] Brian Whitman and Steve Lawrence. Inferring Descriptions and Similarity for Music from Community Metadata. In *Proceedings of ICMC*, 2002.

[44] http://en.wikipedia.org/wiki/Music_charts (access: March 2010).

[45] http://en.wikipedia.org/wiki/List_of_towns_and_cities _with_100,000_or_more_inhabitants/country:_A-B (access: March 2010).

[46] Justin Zobel and Alistair Moffat. Exploring the Similarity Space. *ACM SIGIR Forum*, 32(1):18–34, 1998.