Exploiting Social Media for Music Information Retrieval

Markus Schedl

Abstract This chapter will first provide an introduction to information retrieval (IR) in general, before briefly explaining the research field of music information retrieval (MIR). Hereafter, we will discuss why and how social media mining (SMM) techniques can be beneficially employed in the context of MIR. More precisely, motivations for the common MIR tasks of *music similarity computation*, *music popularity estimation*, and *auto-tagging music* will be provided, and the current state-of-the-art in employing SMM techniques to these three tasks will be elaborated.

Developing music similarity measures is an important task in MIR as such measures are a key ingredient for music recommendation systems, automated playlist generators, and intelligent browsing interfaces, among others. In this chapter, it will be shown how to infer music similarity information from microblogs, collaborative tags, web pages, playlists, and peer-to-peer networks. Estimating the popularity of a music item is obviously important for the music industry but also to create serendipitous music retrieval and recommendation systems. Therefore, approaches that derive such information from web page counts, geo-located microblogs, a peer-to-peer network, and a social music platform will be reviewed. Eventually, different music auto-tagging methods that assign semantic labels to music pieces will be presented. In particular, computational approaches that rely on machine learning techniques as well as human-centred strategies that infer tags directly from some kind of user input (e.g. "games with a purpose") will be addressed.

M. Schedl (✉)
Department of Computational Perception, Johannes Kepler University, Altenberger Straße 69, 4040 Linz, Austria
e-mail: markus.schedl@jku.at

N. Ramzan et al. (eds.), *Social Media Retrieval*, Computer Communications and Networks, DOI 10.1007/978-1-4471-4555-4_20, © Springer-Verlag London 2013

1 Introduction to Information Retrieval

The discipline of information retrieval (IR) is a mature field of research as early work dates back to the 1950s, for instance [59]. Since I can only give a very brief introduction to this exciting field here, the interested reader is referred to one of the many excellent books that offer comprehensive coverage of IR. I personally recommend [22] for an introduction and [3] and [8] for a more comprehensive coverage.

Broadly speaking, IR is concerned with elaborating and testing methods to uncover information from potentially large corpora of text (traditional IR) or (more recently) multimedia, in response to the user's expression of an *information need*. This information need is usually given as a text *query*, the classical example being a user who types in a query string into his or her preferred *search engine*. Texts are most frequently organised in the form of *documents*, although other representations exist. Hence, it is usually also documents which are returned as response to a query to a search engine.

In order to be able to promptly provide search results for millions of queries issued every hour to major search engines, enormous amounts of computational power are required. But of no lesser importance are highly efficient representations of the documents. For this purpose, an *inverted index* is commonly created from the documents. Such an inverted index stores, for each term t, a list of documents in which t occurs or a list of documents and the precise positions of t within each document. The former is referred to as *document-level inverted index*, *record-level inverted index*, *inverted file index*, or just *inverted file*; the latter is typically named *full inverted document index*, *word-level inverted index*, *full inverted index*, or *inverted list*. The major advantage of a full inverted index is that it allows for *phrase search*, that is, finding an exact phrase within a document, not only a single term. In a regular expression notation, the two variants of the mapping implemented by the two flavours of indexes can be written as follows:

document-level index:	`term`$\mapsto$`document*`
world-level index:	`term`$\mapsto$`(document,position+)*`

If the user now wants to search for a particular topic, expressed as a query q, the retrieval system computes a matching score between q and the indexed documents D. A common approach is to compute term weights $w(q, d)$ between q and each document d, which estimate the importance of the document for the query. The documents are then ranked with respect to $w(q, d)$ and displayed to the user in descending order of term weight. This classical retrieval approach is often called *term vector model* or *vector space model*. Since its proposal in 1975 by Salton et al. [71], many extensions as well as alternative retrieval approaches have been suggested. More recent methods include *probabilistic retrieval* [45] and *graph-based models* [7].

2 Music Information Retrieval at a Glance

Unlike traditional IR, music information retrieval (MIR) is a relatively young field of research, dating back only about a decade. An early and quite general definition of MIR, which highlights the multidisciplinarity of the field, is given by Downie in [17]:

> MIR is a *multidisciplinary* research endeavour that strives to develop innovative *content-based searching schemes*, *novel interfaces*, and *evolving networked delivery mechanisms* in an effort to make the worlds vast store of music accessible to all.

A later definition given by Schedl [72] focuses on extracting and processing musical information on different levels and modalities:

> MIR is concerned with the *extraction*, *analysis*, and *usage* of information about any kind of music entity (for example, a song or a music artist) on any representation level (for example, audio signal, symbolic MIDI representation of a piece of music, or name of a music artist).

Due to recent developments, such as audio and music streaming services (e.g. `Spotify` [41]), personalised web radio (e.g. `last.fm` [27]), and increasing use of multimedia data in social media, MIR has gained considerably in importance as a research field.

Although MIR is a highly multidisciplinary research field, including areas as diverse as music theory, library science, psychology, law, and artificial intelligence, one of its key goals is to better understand how humans perceive, create, process, and interact with music. Given its strong connection to computer science, MIR approaches to achieve this broad goal typically involve elaborating computational models of music perception. These approaches commonly take as input the audio signal or other modalities of a music item and compute *features* that strive to describe particular aspects of the music item, for example, rhythm, harmony, or timbre. Figure 1 depicts a schematic and simplified illustration of how a signal-based (content-based) audio feature extractor works. First, the audio signal is sampled and digitised, yielding a representation as *pulse code modulation* (PCM). For instance, when producing a compact disc, the sampling frequency is typically 44,100 Hz, and each sample is described via 16 bits. For a stereo recording, the data volume hence amounts to 176,400 bytes per second. The PCM representation is then split into (often overlapping) frames with a typical length of between 2^8 and 2^{12} samples. Low-level features in the *time domain* can then be computed directly on these frames. To capture frequency information, alternatively, it is very common to apply a *windowing function* to each frame and subsequently compute the *fast Fourier transform* (FFT) [13], which converts the data from a time–amplitude representation into a frequency–magnitude system. Hereafter, several post-processing steps are commonly performed, for instance, employing some psychoacoustic model of human auditory perception. Eventually, one regularly has to decide how to combine the features computed for each frame of a piece of music to create a global representation. Methods range from computing simple statistical moments to complex time-series modelling via *hidden Markov models* (HMM) [5].

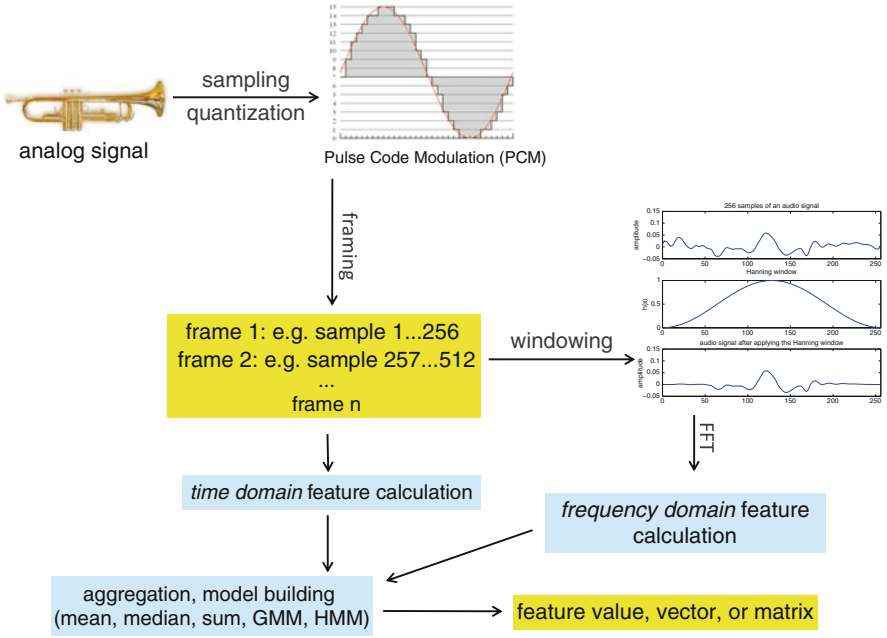

Fig. 1 Basic scheme of an acoustic feature extractor

The computational features extracted via algorithms similar to the one just described can be used for a wide range of MIR tasks, for instance, to *estimate similarity between music items* which in turn enables the creation of *music recommendation systems*, of *playlists automatically generated*, and of clustering-based *user interfaces* to music collections. If semantic labels describing the music items are available, another popular task is to automatically learn relations between audio features and semantic descriptors. This task is commonly referred to as *auto-tagging*.

The content-based feature extraction framework described above represents the traditional MIR strategy to computationally grasp aspects of a music item that should relate to human music perception. In the past few years, however, MIR has seen a paradigm shift to incorporate additional factors into computational models of music perception and description. In particular, contextual aspects of the music items and of the listener are increasingly taken into account. Integrating these with traditional content-based methods, Fig. 2 shows the three broad pillars from which perceptual music information can be extracted, according to [76].

Music content feature extractors derive information directly from the audio representation of a piece of music, by applying signal processing techniques. A typical example are features inferred from time-invariant *Mel frequency cepstral coefficients* (MFCC) representations of the audio signal, which serve to some extent to describe the coarse timbre of an audio signal. Overviews of common content-based extraction techniques are provided, for instance, in [9, 20, 57].

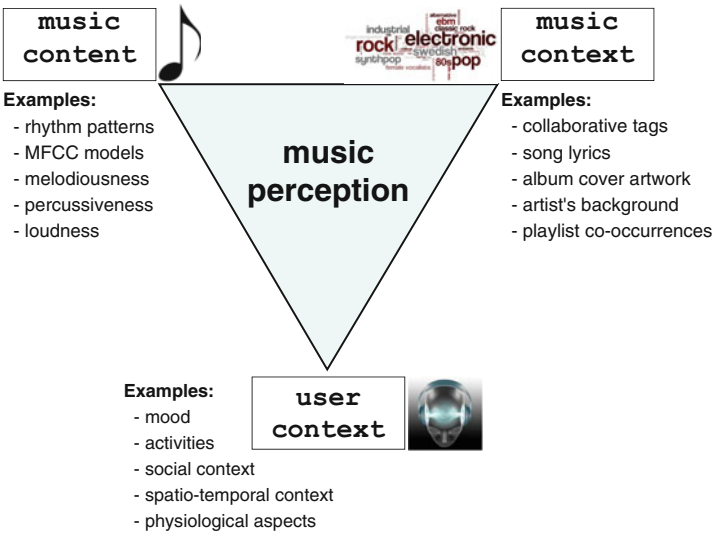

Fig. 2 Categorisation of computational aspects that influence music perception

Music context refers to aspects that are not encoded in the audio signal (or cannot be extracted with current methods), nevertheless are related to a music item. For instance, collaborative tags about a performer, semantic meaning of song lyrics, or the political background of an artist fall into this category. More details on feature extraction and similarity estimation from the music context can be found, for example, in [75].

User context relates to personal properties, preferences, and feelings of the music listener. The user context hence includes the user's mood, activities, friends, or level of musical training. Although these highly individual factors are obviously influential on music perception, MIR literature centred around the user is relatively sparse. Among the existing work, I would like to highlight the following: Cunningham et al. present an interesting study on why people dislike particular music [15]; Lee conducted a thorough analysis of natural language music queries [55] and personalised and user-aware music retrieval and recommendation are treated, for example, in [4, 10, 81].

Finally, it is noteworthy that some aspects fall into more than one category. For example, song lyrics might be seen to belong to the *music content* as they are obviously encoded in the audio signal. However, with current MIR techniques, it is impossible to extract and convert them to a semantically meaningful textual representation. On the other hand, many web pages list huge amounts of song lyrics, which make it easy to extract them from a *contextual* data source. I therefore predominantly see them in the *music context* category. A similar overlap might occur for collaborative tags. One can argue that such tags are the outcome of many users, hence would count them to the *user context*. However, according to my categorisation, the *user context* refers to individual, personal factors of the user, not to user groups.

3 Social Media Mining in Music Information Retrieval

Usage of social media has seen a tremendous increase during the past couple of years. People create, modify, and most importantly share massive amounts of multimedia data (text, images, music, videos) on platforms such as `Twitter` [42], `Facebook` [34], `last.fm` [27], and `YouTube` [43].

As music plays a vital role in many human lives and everyone has an opinion about music, user-generated content related to music items such as artists, performers, songs, albums, or music videos is available in abundance. Given the remarkable commercial interest in music distribution and delivery, innovative music retrieval systems are becoming increasingly important. Such systems include personalised, user-aware music recommenders [4], automated playlist generators [64], or intelligent browsing interfaces [48] that transcend the traditional filtering-based browsing scheme according to an artist–album–track hierarchy.

Given the huge amount of user-generated data and the broad interest in music, elaborating sophisticated methods to mine social media content in order to derive semantic information about music and other media items is an ongoing research endeavour, which is currently pursued quite actively. In the following, we will hence discuss the state of the art in three key areas of MIR, where social media mining (SMM) can help improve upon traditional solutions. More precisely, the topics covered are how to compute similarities between music items such as songs or artists, how to estimate the popularity of a music item, and how to tag music items, that is, assign semantic labels to a piece of music, album, or artist.

3.1 Music Similarity Estimation

Computing similarity estimates between two music items (e.g. songs or artists) is an important task in MIR as it enables, among others, automated creation of playlists, recommending items similar to the favourites of a user, or applying clustering techniques and consequently creating user interfaces that foster browsing music collections in an intuitive way.

An example for automated music playlist generation is [68], where content-based data and contextual data (extracted from music-related web pages) are combined to create seamless playlists. Pohle et al. aim at creating playlists in which consecutive tracks sound as similar as possible. Figure 3 shows a music browser entitled `Traveller's Sound Player`, which allows to interact with the generated playlists.

A user interface to music collections, named `nepTune`, is presented in [48], where Knees et al. extract audio features from digital audio files to train a *self-organising map* (SOM) [51]. The SOM uses similarities between feature representations of songs to cluster the music collection under consideration. The clusters are then visualised via first estimating the distribution of the data items over the map and subsequently using the estimated densities as height values to create a

Fig. 3 Screenshot of the `Traveller's Sound Player` interface for automated playlist generation

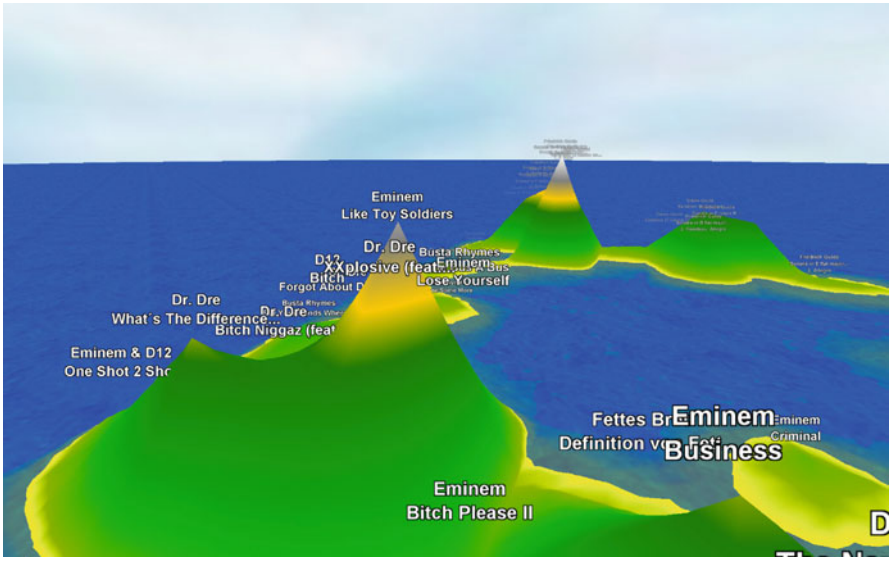

Fig. 4 Screenshot of the `nepTune` browsing interface for music collections

virtual landscape of the music collection. The landscape generated in this way can then be navigated through in the manner of a computer game. Figure 4 shows a screenshot of the `nepTune` interface.

Various kinds of social media have been used to derive similarity scores between music items. In the following, we will particularly focus on methods that construct a similarity measure from user-generated shared *playlists* (e.g. available from `Art of the Mix` [30]) [2], *shared folders in P2P networks* [58], *microblogs* [80], and *collaborative tags* [19,56]. Social media sources for collaborative tags include dedicated platforms such as `last.fm` or the recently quite popular "games with a purpose" [54,61,90].

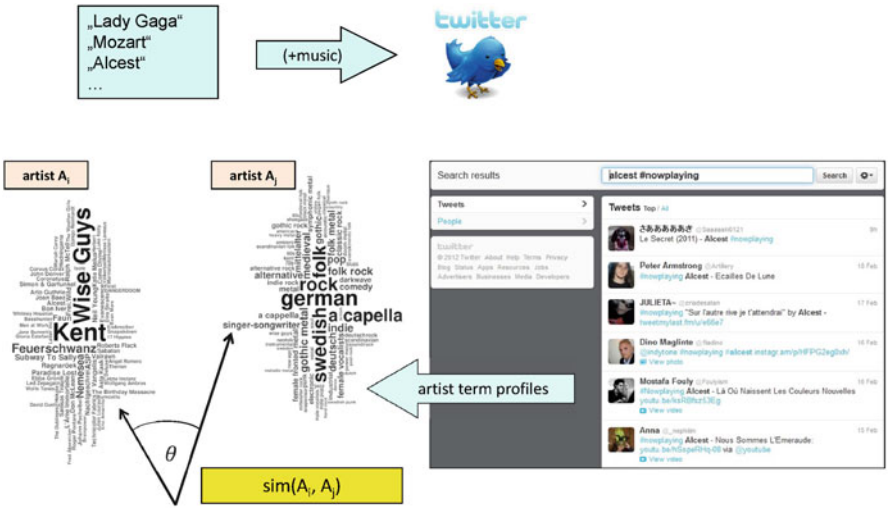

Fig. 5 Artist similarity estimation from microblogs

According to the exploited data source and similarity computation strategy, the methods under discussion can be categorised into *text-based* (microblogs and collaborative tags) and *co-occurrence-based* (web pages, playlists, and shared folders in P2P networks), each of which requires different algorithms to construct a similarity measure. Evaluation, on the other hand, can be performed using the same techniques; most common are *genre classification* and comparison against *human similarity judgements*.

3.1.1 Text: Microblogs and Collaborative Tags

Text-based approaches to music similarity estimation typically approximate the similarity by employing the vector space model, which was introduced in Sect. 1. In the following, we will discuss how to derive similarity information from microblogs and from collaborative tags extracted from `last.fm` or gathered from “games with a purpose”.

Microblogs

A comprehensive study of different aspects in estimating music artist similarity from microblogs is presented in [74]. For the experiments in this chapter, the *vector space model* was applied, that is, each artist is modelled as a *term weight vector* in a high-dimensional *feature space* and similarities between these term vector representations are calculated. An overview of the basic approach is depicted in Fig. 5. First, the `Twitter` API is used to retrieve microblogs for

each artist in a given list of 3,000 music artists. The returned tweets for each artist are then concatenated, resulting in a *virtual artist document*, and a term vector representation for each artist is computed. The actual similarity estimate between two artists A_i and A_j is eventually obtained by calculating a similarity1 function S_{A_i,A_j}.

In the study presented in [74], several thousand combinations of the following single aspects have been assessed:

- Query scheme
- Index term set
- Term frequency (TF)
- Inverse document frequency (IDF)
- Normalisation with respect to document length
- Similarity function

Evaluating different *query schemes* is motivated by the fact that earlier work in web-based MIR has shown an improvement in the accuracy of similarity estimates when adding music-related keywords to the search query (e.g. "music" or "music review") [47, 78, 92]. Different *index term sets*, that is, lists of terms used to filter the microblogs and create the term weight vectors, have been assessed as well. The number of terms in the index term set corresponds to the dimensionality of the respective feature vectors (TF · IDF vectors). The *term frequency* $r_{d,t}$ of a term t in a virtual artist document d estimates the importance t has for document d, hence for the artist under consideration. The *inverse document frequency* w_t estimates the overall importance of term t in the whole corpus and is commonly used to weight the $r_{d,t}$ factor, that is, downweight terms that are important for many documents and hence less discriminative for d. Performing this calculation for all terms in the used index term set and each virtual artist document results in one TF · IDF vector per artist. It is common to subsequently *normalise* the TF · IDF vectors with respect to document length. Finally, different *similarity functions* S_{d_i,d_j} to estimate the proximity between the term vectors of two virtual artist documents d_i and d_j are examined.

As for evaluation, *mean average precision* (MAP) scores are computed on genre labels predicted by various classifiers. More precisely, given a query or seed artist, the retrieval task is to find artists of the same genre via similarity. MAP is simply computed as the arithmetic mean of the $precision@k$ scores, that is, the average precision of k-nearest neighbour (kNN) classifiers for varying values of k.

Although reporting all results of [74] is way beyond the scope of this chapter, I would like to summarise the most important findings in the following.

Query Scheme: When dealing with microblogs, it is preferable to use only the artist name (no additional keywords) to query the `Twitter` API or more general to select the tweets relevant to the artist under consideration.

Index Term Sets: Even though using all terms in the corpus yields the highest MAP values, results are by far the most unstable ones. This means that slightly modifying a single other aspect can cause a significant decline in accuracy when

using all terms in the corpus. Given the high computational complexity due to feature spaces of dimensionality greater than one million, employing no particular index term set is not favourable. Best and most robust results were achieved on average using a dictionary of musical genres, musical instruments, and emotions, which was gathered from `Freebase` [35].

Term Frequency: A simple binary match TF formulation should not be used. The most favourable algorithmic variants are logarithmic formulations and an adapted *Okapi BM25* formulation [69,70].

Inverse Document Frequency: Among the IDF formulations, binary match yields the worst results. Also signal estimates and signal-to-noise ratios do not perform much better. Again, logarithmic formulations and the modified *Okapi BM25* formulation yield top results.

Normalisation: Performing no normalisation for document length performs best, both in terms of accuracy and robustness. This is presumably due to the special characteristics of tweets, which are limited to 140 characters, a limit commonly exhausted by `Twitter` users. Normalisation hence does not improve results but increases computational costs.

Similarity Function: Among the similarity functions under estimation, the *Jeffrey divergence*-based function performs very well, while at the same time maintaining a reasonable stability level. Also the *Jaccard coefficient* performs remarkably well. *Euclidean similarity* performed inferior in all combinations.

Overall, the best performing variants found in the experiments are given by the three term-weighting functions in Eqs. 1–3, in combination with the *Jaccard coefficient* similarity function (Eq. 4). In these equations, N represents the total number of documents in the corpus, $f_{d,t}$ is the number of occurrences of term t in document d, f_t denominates the total number of documents containing term t, W_d is the document length of d, and $\mathcal{T}_{d_1,d_2}$ denotes the set of distinct terms in documents d_1 and d_2.

$$w_{d,t} = \log_e(1 + f_{d,t}) \cdot \log_e \frac{N - f_t}{f_t} \tag{1}$$

$$w_{d,t} = \log_e(1 + f_{d,t}) \cdot \log \frac{N - f_t + 0.5}{f_t + 0.5} \tag{2}$$

$$w_{d,t} = (1 + \log_e f_{d,t}) \cdot \log_e \frac{N - f_t}{f_t} \tag{3}$$

$$\text{sim}(d_1, d_2) = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} (w_{d_1,t} \cdot w_{d_2,t})}{W_{d_1}^2 + W_{d_2}^2 - \sum_{t \in \mathcal{T}_{d_1,d_2}} (w_{d_1,t} \cdot w_{d_2,t})} \tag{4}$$

Table 1 Most popular tags and their artist frequencies, among a set of 1,995 artists

Tag	Frequency
Jazz	809
Seen live	658
Rock	633
60s	623
Blues	497
Soul	423
Classic rock	415
Alternative	397
Funk	388
Pop	381
Favourites	349
American	345
Metal	334
Electronic	310
Indie	309

Table 2 Tags assigned by `last.fm` users only once, among a set of 1,995 artists

Crappy girl singers
Stuff that needs further exploration
Disco noir
Knarz
Lektroluv compilation
Gdo02
Electro techo
808 state
Good gym music
Techno manchester electronic acid house
Music i tried but didnt like
American virgin festival

Collaborative Tags

User-generated tags that are assigned to music items are a valuable, albeit noisy source for different MIR tasks, not least for similarity estimation and music retrieval purposes.

Geleijnse et al. gather tags from `last.fm` to generate a "tag ground truth" on the artist level [19]. The authors first filter redundant and noisy tags using the set of tags associated with tracks by the artist under consideration. Similarity between two artists is then estimated as the number of overlapping tags. Evaluation on a set of 1,995 artists, using `last.fm`'s similar artist function as ground truth, shows that the number of overlapping tags between similar artists is much larger than the overlap between arbitrary artists (about 10 vs. 4 tags after filtering). Another interesting observation is that the tags assigned to the largest number of artists fall into only three semantic categories – genres, personal references, and time periods (Table 1). The least frequent tags are shown in Table 2. Often they are more prosaic, represent specific personal notes, or simply contain typos.

Another work on collaborative tags is [56], where Levy and Sandler construct a semantic space for music pieces based on tags retrieved from `last.fm` and `MusicStrands` [28], a web service (no longer in operation) that allowed users to share playlists. To this end, all tags found for a specific music piece are tokenised, and a document-term matrix based on TF · IDF weighting is created. Each track is hence represented by a term vector. For the TF part of the weighting, three different approaches are considered: using the number of users that applied the tag, ignoring the number of users (performing no TF weighting at all), and restricting the terms to adjectives by employing a part-of-speech (POS) tagger. Levy and Sandler further analyse the influence of applying *latent semantic analysis* (LSA) [16] to reduce the dimensionality of the feature space. The authors then compute the similarity between the resulting feature vectors using the cosine measure. For evaluation, the authors employ a retrieval scenario and report *average precision* values. They judge the relevance of retrieved terms as having assigned the same genre or artist label as the seed. Levy and Sandler find that using all terms (not only adjectives) is preferable. They also found the incorporation of the number of users that applied the tag into the TF score superior.

It was in 2007 when the MIR community recognised the value of "games with a purpose" for MIR tasks. In this very year, three papers proposing different music-tagging games were found in the proceedings of the annual "International Society for Music Information Retrieval" (ISMIR) conference, the main scientific venue for MIR research. The principal motivation for such games is to let users solve tasks that are hard or even infeasible to perform for a computer, while at the same time being entertaining enough to attract and keep many users playing. In the music domain, Law et al. present `TagATune`, a game for semantic annotation of music and audio [54]. Two players are paired and are then listening to the same piece of audio. They can describe the audio by entering words but are rather told to guess what their partners are thinking, because both players will only score points if their tags match. If this is the case for one tag, the game will proceed to the next track. Even though `TagATune` was originally designed to harvest semantic descriptions for music and audio, it also implements a "comparison round", where users are presented three songs – one seed track and two alternatives to choose from. They then have to decide which of the alternatives sound more similar to the seed song. From this kind of information, relative similarity judgements and in turn a similarity measure can be derived, as done by Law and von Ahn [53], Stober [88], and Wolff and Weyde [93], for instance.

A similar game, called `Listen Game`, is presented by Turnbull et al. in [90]. It aims at uncovering semantic relationships between words and music. Again, players are grouped and listen to the same songs. They subsequently have to choose from a list of words the one that best and the one that worst describes the song. Users get immediate feedback about which tags other players have chosen. From the data collected, Turnbull et al. employ the *mixture hierarchies expectation maximisation* (MH-EM) [91] algorithm to learn semantic associations between words and songs. These associations are weighted and can therefore be used to construct tag weight vectors for songs and in turn to define a similarity measure for retrieval [89].

Table 3 Most popular tags assigned by players of a "game with a purpose" on music annotation

Tag	Frequency
Drums	793
Guitar	720
Male	615
Rock	571
Synth	429
Electronic	414
Pop	375
Bass	363
Female	311
Dance	297
Techno	224
Electronica	155
Piano	153
Rap	140
Synthesizer	136

Mandel and Ellis present another game for music annotation in [61]. It differs from the other games presented so far in that it uses a more fine-grained scoring scheme. Players receive more points for new tags to stimulate the creation of a larger semantic corpus. More precisely, a player who first uses a tag t to describe a particular song scores two points if t is later confirmed (used again) by another player. The third and subsequent players that use the same tag t do not receive any points. Thus, players who are the first to use a word t for tagging a particular song do not receive an immediate reward but will score two points as soon as another player will have used t. The authors report the most popular tags confirmed by at least one user. They are summarised in Table 3. Compared to the top tags extracted from `last.fm` (Table 1), the tags originating from the tagging game more often describe instruments and gender of the main performer.

3.1.2 Co-occurrences: Web Pages, Playlists, and P2P Networks

The family of *co-occurrence approaches* to music similarity estimation is based upon the assumption that *two music items are more likely to be similar if they co-occur in the same document*, for instance, a playlist, a web page, or a tweet.

Web Pages

In this vein, [78] defines the similarity of two artists as the conditional probability that one artist is to be found on a web page that is known to mention the other artist. This conditional probability can either be calculated on *crawled web pages* that relate to the artists under consideration or heuristically approximated using *page count information* from major search engines.

The former strategy, performing web crawls to infer similarities, is followed in [12] and [72][Chap. 3]. To this end, a certain amount of top-ranked web pages returned by a search engine is retrieved for each artist A_i. Subsequently, all pages fetched for A_i are searched for occurrences of all other artist names in the collection. The number of page hits represents a co-occurrence count that equals the document frequency of the artist term "A_j" in the corpus of web pages for artist A_i. This count is expressed by the asymmetric function $\text{cooc}(A_i, A_j)$. A similarity score is then computed by relating this count to the total number of pages successfully fetched for artist A_i. Symmetrising these scores for all pairs of artists eventually leads to the similarity function shown in Eq. 5. Please note that $\text{cooc}(A_i, A_i)$ and $\text{cooc}(A_j, A_j)$ refer to the total number of web pages successfully crawled for artists A_i and A_j, respectively.

$$\text{sim}(A_i, A_j) = \frac{1}{2} \cdot \left[\frac{\text{cooc}(A_i, A_j)}{\text{cooc}(A_i, A_i)} + \frac{\text{cooc}(A_j, A_i)}{\text{cooc}(A_j, A_j)} \right] \tag{5}$$

The heuristic solution referred to in the beginning of this section is proposed in [78]. It relies solely on the page count estimates provided by a search engine. In short, these page count estimates for queries like `"artist name i"` or `"artist name i"+"artist name j"` are used to infer the relative frequency of both artists' co-occurrence and in turn the conditional probability as indicated above. Equation 6 gives a formal representation of the symmetrised similarity function.

$$\text{sim}(A_i, A_j) = \frac{1}{2} \cdot \left[\frac{pc(A_i, A_j)}{pc(A_i)} + \frac{pc(A_i, A_j)}{pc(A_j)} \right] \tag{6}$$

Comparing the two strategies (web crawls and page count estimates) in terms of computational complexity, it is obvious that the former one requires fewer requests to the search engine. The number of queries to the search engine grows indeed linearly with the number of music items in the collection. In contrast, the second approach that entirely relies on page count estimates from search engines grows quadratically in the number of queries. It is hence less suited for mid- and large-size music collections.

Playlists

Exploiting co-occurrence information from playlists to derive a similarity estimate between music items was probably first suggested in [66]. Pachet et al. consider radio station playlists from a French radio channel and compilation CDs from

CDDB[1] to extract co-occurrences between tracks and between artists. The authors count the number of co-occurrences of two artists (or pieces of music) A_i and A_j in the radio station playlists and compilation CDs. They define the co-occurrence of an entity A_i to itself as the number of A_i's occurrences in the considered data source. To account for different frequencies, that is, popularities, of songs or artists, the co-occurrence counts are normalised. Assuming that co-occurrence is a symmetric function, the similarity measure used by the authors is the same as given1 by Eq. 5.

Focusing on social media data, Baccigalupo et al. present an approach to derive artist similarity information from playlists shared by members of a web community [2]. The authors look at more than one million playlists made publicly available by MusicStrands [28]. The authors extract the 4,000 most popular artists from the playlist set, measuring popularity as the number of playlists in which each artist occurs. They further take into account that two artists consecutively occurring in a playlist are probably more similar than two artists occurring farther away in a playlist. To this end, the authors define a distance function $d_h(A_i, A_j)$ that counts how often a song by artist A_i co-occurs with a song by A_j at a distance of h. Thus, h is a parameter that reflects the number of songs in between the occurrence of a song by A_i and the occurrence of a song by A_j in the same playlist. The distance between two artists A_i and A_j is defined by Eq. 7, where the playlist counts at distances 0 (two consecutive songs by artists A_i and A_j), 1, and 2 are weighted with factors β_0, β_1, and β_2, respectively. The authors empirically set the weights to $\beta_0 = 1$, $\beta_1 = 0.8$, and $\beta_2 = 0.64$.

$$\mathrm{dist}(A_i, A_j) = \sum_{h=0}^{2} \beta_h \cdot \left[d_h(A_i, A_j) + d_h(A_j, A_i) \right] \tag{7}$$

$$|\mathrm{dist}|\,(A_i, A_j) = \frac{\mathrm{dist}(A_i, A_j) - \widehat{\mathrm{dist}}(A_i)}{\left| \max\left(\mathrm{dist}(A_i, A_j) - \widehat{\mathrm{dist}}(A_i) \right) \right|} \tag{8}$$

$$\widehat{\mathrm{dist}}(A_i) = \frac{1}{n-1} \cdot \sum_{j \in X} \mathrm{dist}(A_i, A_j) \tag{9}$$

To account for the *popularity bias*, that is, very popular artists co-occur with a lot of other artists in many playlists simply because they are well known and often listened to by the average music listener, the authors perform normalisation according to Eq. 8, where $\widehat{\mathrm{dist}}(A_i)$ denotes the average distance between A_i and all other artists (Eq. 9) and X is the set of the $n-1$ artists other than A_i.

[1] CDDB is a web-based album identification service that returns, for a given unique disc identifier, meta-data like artist and album name, tracklist, or release year. This service is offered in a commercial version operated by Gracenote [38] as well as in an open source implementation named freeDB [36].

Peer-to-Peer Networks

Peer-to-peer (P2P) networks represent another source of music-related data since users of this kind of network are commonly willing to reveal meta-data about their shared content. For music files, meta-data typically shared is filenames and ID3 tags. By analysing which items co-occur in a user's shared folder, researchers have created music similarity measures.

Among early work that makes use of data extracted from P2P networks is [18, 58, 92], and [6]. These papers all extract data from the P2P network `OpenNap` to derive music similarity information.[2]

Logan et al. [58] and Berenzweig et al. [6] report on having determined the 400 most popular artists on `OpenNap` in mid-2002. The authors harvested meta-data on shared content, which yielded about 175,000 user-to-artist relations from about 3,200 shared music collections. Logan et al. [58] especially highlights the sparsity in the `OpenNap` data, in comparison with music content data. Logan et al. compare similarities defined by artist co-occurrences in `OpenNap` collections, by expert opinions from `allmusic.com` [29], by playlist co-occurrences from `Art of the Mix`, by data gathered from a web survey, and by audio feature extraction (MFCCs) [1]. They calculate a "ranking agreement score" by comparing the top N most similar artists according to each data source and calculating the pairwise overlap between the sources. Their 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 agree moderately, and the signal-based measure has a rather low agreement with all other sources (except for comparison to the allmusic.com data).

Whitman and Lawrence use a software agent to retrieve from `OpenNap` a total of 1.6 million user–song relations over a period of 3 weeks in August 2001 [92]. To alleviate the *popularity bias*, the authors use a similarity measure as shown in Eq. 10, where $C(A_i)$ denotes the number of users that share songs by artist A_i, $C(A_i, A_j)$ is the number of users that have both artists A_i and A_j in their shared collection, and A_k is the most popular artist of the whole data set. The second factor (in the right-hand part of the equation) downweights the similarity between two artists if one of them is very popular and the other is not.

$$\text{sim}(A_i, A_j) = \frac{C(A_i, A_j)}{C(A_j)} \cdot \left(1 - \frac{\left|C(A_i) - C(A_j)\right|}{C(A_k)}\right) \tag{10}$$

In [18], Ellis et al. aim to build a ground truth for artist similarity estimation. The authors report on having extracted from `OpenNap` about 400,000 user-to-song relations, covering about 3,000 unique artists. Again, the co-occurrence

[2] It is not clear whether the four mentioned publications make use of exactly the same data set. In any case, the authors emphasise that they only extract meta-data from `OpenNap`, but do not download any files.

data is compared with artist similarity data gathered by a web survey and with `allmusic.com` data. In contrast to Whitman and Lawrence, Ellis et al. take indirect links in `allmusic.com`'s similarity judgements into account. To this end, Ellis et al. propose a transitive similarity function on similar artists from the `allmusic.com` data, called "Erdös distance". More precisely, the distance $d(A_i, A_j)$ between two artists A_i and A_j is measured as the minimum number of intermediate artists needed to form a path from A_i to A_j. As this definition also allows to derive information on dissimilar artists (with a high minimum path length), it can be employed to obtain a complete distance matrix.

A recent approach by Shavitt and Weinsberg derives similarity information on the artist and on the song level from the `Gnutella` file-sharing network [84]. The authors collected meta-data of shared files from more than 1.2 million `Gnutella` users in November 2007. They restricted their search to music files (MP3 and WAV). The crawl yielded a data set of 530,000 songs. Information on both users and songs are represented via a 2-mode graph showing users and songs. A link between a song and a user is created if the user shares the song. Analysing the resulting network, it turned out that most users of the P2P network share similar files.

Shavitt and Weinsberg further propose an approach to *artist recommendation*. To this end, they construct a user-to-artist matrix V, where $V(i, j)$ gives the number of songs by artist A_j that user U_i shares. They subsequently perform direct clustering on V using the k-means algorithm [60] with the Euclidean distance metric. Artist recommendation is then performed using either data from the centroid of the cluster to which the seed user U_i belongs or information about the nearest neighbours of U_i within the cluster to which U_i belongs.

In addition, Shavitt and Weinsberg address the problem of *song clustering*. Accounting for the *popularity bias*, the authors define a distance function that is normalised according to song popularity, as shown in Eq. 11, where $uc(S_i, S_j)$ denotes the total number of users that share songs S_i and S_j. C_i and C_j denote, respectively, the popularity of songs S_i and S_j, measured as their total occurrence in the data set:

$$\text{dist}(S_i, S_j) = -\log_2\left(\frac{uc(S_i, S_j)}{\sqrt{C_i \cdot C_j}}\right) \tag{11}$$

3.2 Music Popularity Estimation

Estimating the popularity of a music artist or song in a certain region of the world is an important task, not only for the music industry. Also the cosmopolitan and culturally aware music aficionado is likely to be interested in which music is currently "hot" in different parts of the world. Not least artists are interested to know where in the world their music is particularly (un)popular. Furthermore, popularity information can serve as an important component for *serendipitous music retrieval* systems [10, 81].

An artist's or song's popularity can be estimated via a wide variety of predictors, such as traditional charts (e.g. "Billboard Hot 100" released weekly for the United States of America by the Billboard Magazine [26]), microblogging activity, playcounts (e.g. from last.fm or YouTube), occurrences on web pages, and shared folder analysis in P2P networks.

Scientific work on this topic includes [73], where Schedl et al. compare different data sources for artist popularity estimation on a per-country basis. In [50], Koenigstein et al. analyse search queries issued within a P2P network to infer music popularity. Grace et al. compute popularity rankings from user comments in a social network [21].

Given the large interest record companies, producers, and artists have in this kind of information, it is not surprising that there also exist businesses specialised on music popularity measurement. Examples are Band Metrics [31] or BigChampagne Media Measurement [32]. Even though they obviously do not reveal details of their algorithms, it can be reasonably assumed that these companies harvest multiple data sources to create their predictors. The music information platform Echo Nest [25] even offers a public API function to retrieve a ranking based on the so-called "hottness" of an artist [24]. This ranking is based on editorial, social, and mainstream aspects [23]. However, this web service does not provide country-specific information, and Echo Nest is known to have a strong focus on the USA.

In the following, approaches that make use of social media to predict the popularity of an artist or a song will be presented and discussed. Also properties of the data sources, such as particular biases, availability, noisiness, and time dependence, will be addressed.

3.2.1 Data Sources for Popularity Estimation

The popularity of an artist or track can be defined on different levels of granularity (e.g. individual user, peer group, country, or cultural region). Incorporating previous approaches presented in [49, 50], Schedl et al. compare different ways to derive popularity information from various social media sources [79] on the level of countries. To this end, a framework is established that uses the following proxies for popularity:

- Page counts of web pages
- Artist occurrences in geo-located microblogs
- Meta-data from folders shared in the Gnutella P2P network
- Playcount data from the social music platform last.fm

The approaches proposed to compute popularity rankings from each data source are detailed below.

Another work that infers popularity information from social media is [21], where Grace et al. compute popularity rankings from user comments in the social network MySpace [40]. 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 Billboard charts.

Web Page Counts

Page counts are gathered by querying the web search engines Google [37] and Exalead [33] for ⟨artist, country⟩ tuples. To guide the search towards musically relevant web pages and avoid distortions caused by artist names that equal common speech words (e.g. "Bush", "Kiss", "Hole"), the query scheme `"artist name" "country name" music` is employed. Furthermore, a factor resembling *inverse document frequency* (IDF) is used to downweight popularity of artists that are popular everywhere in the world since the aim is to uncover popular artists specific to each country. The final ranking score is calculated according to Eq. 12, where $pc_{c,a}$ is the page count value returned for the country-specific query for artist a and country c, $|C|$ 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 $pc_{c,a} > 0$).

$$\text{popularity}_{c,a} = pc_{c,a} \cdot \log_2\left(1 + \frac{|C|}{df_a}\right) \tag{12}$$

Geo-Located Microblogs

Microblogs are retrieved from Twitter using the search API and are then narrowed in two ways. First, only posts containing the hashtag #nowplaying are considered. This filtering is directly supported by the Twitter API. Secondly, the search is narrowed to a specific country. To this end, posts are categorised according to their location within a certain radius around the major cities of the world. Tweets are then aggregated to the country level. Scanning the retrieved microblogs for occurrences of the artists of interests and counting the number of their appearances for a given country c eventually yield a count equal to the term frequency ($tf_{c,a}$) of artist a in an aggregated document comprising all tweets gathered for cities in country c. Equation12 again gives the ranking score, when $pc_{c,a}$ is replaced with $tf_{c,a}$.

P2P Network

Shared folder data from the P2P network Gnutella is extracted employing a two-stage process, similar to [49]: a *crawler* component discovers the highly dynamic

network topology; a *browser* queries the active nodes – corresponding to users – for meta-data of files in their shared folders. The crawler treats the network as a graph and performs *breadth-first exploration*. Discovered active nodes are enqueued in a list that is processed by the browser. Shared digital content is associated with artists by matching the artist names of interest against ID3 tags of shared music files. Occasionally ID3 tags are missing or misspelled. Artists names are therefore also matched against the filenames. Creating popularity charts for specific countries requires determining the geographical location of the users. The necessary geo-identification process is based on IP addresses. First, a list of all unique IP addresses in the data set – typically over a million – is created. IP addresses are then geo-located using the commercial `IP2Location` [39] database. Each IP address is hence attached a country code, a city name, and latitude–longitude coordinates. The geographical information obtained in this way pinpoints fans and enables tracking spatial diffusion of artists popularity [50]. Aggregating the amount of digital content associated with each artist for the country under consideration yields the final ranking score.

Social Music Platform

As last data source, artist popularity based on the user community of the social music platform `last.fm` is considered. Despite the issues of *hacking and vandalism* and a certain *community bias* [75], which are inherent to collaborative music information systems, the playcounts of `last.fm` users can be expected to reflect which music is currently popular in this community. First, the top 400 listeners of each country are gathered via the `last.fm` API. The most frequently played artists for each of these listeners are extracted subsequently.[3] Aggregating these playcounts for each ⟨artist, country⟩ pair finally yields a popularity ranking.

3.2.2 A Multifaceted Comparison of Different Data Sources

It was shown in [79] that the popularity charts obtained from the different, inhomogeneous data sources do not correlate highly. Each data source hence covers different aspects of popularity, which indicates that the quest for artist popularity is a multifaceted and challenging task, especially in the era of multichannel music distribution.

Trying to uncover the different dimensions of the five data sources (the four web and social media sources and traditional music charts), Table 4 compares

[3] In the meantime, `last.fm` has extended its API with a `Geo.getTopArtists` function that returns the top-played artists in a particular country.

Table 4 Comparing different social media sources to infer popularity information

Source/aspect	Bias	Availability	Noisiness	Time dependence
Web page counts	Web users	Widespread	High	Accumulating
Twitter	Community	Country-dependent	Medium	Current
P2P	Community	Country-dependent	Low–medium	Accumulating
Last.fm	Community	Widespread	Medium–high	Accumulating
Traditional charts	Music industry	Country-dependent	Low	Current

Table 5 Availability of data for popularity estimation

Data source	Countries
Web page counts	240
Twitter	155
P2P	86
Last.fm	240

them according to several criteria relevant to the task of popularity estimation. One issue is that certain approaches are prone to a specific *bias*. The average `last.fm` user, for instance, does not represent the average music listener of a country, that is, `last.fm` data is distorted by a *community bias*. The same holds for `Twitter`, which is biased towards artists with very active fans. On the other hand, some very popular artists may have fans who use `Twitter` to a much lower degree. Traditional charts are frequently biased towards the record sales figures the music industry commonly uses as proxy.

Another aspect is data *availability*. While page count estimates are available for all countries of the world, the approaches based on P2P and `Twitter` data suffer from a very unbalanced coverage for different countries. Also traditional music charts vary strongly in terms of availability between countries. Table 5 shows the number of countries for which data could be extracted for each approach, as presented in [79]. Please note that these results are based on a list of 240 countries retrieved from `last.fm`.

A big advantage of traditional charts is their robustness against noise. In contrast, *page count estimates* are easily distorted by ambiguous artist or country names. `Last.fm` data suffers from hacking and vandalism [11], as well as from unintentional input of wrong information and misspellings.

According to the dimension of *time dependence*, the data sources can be categorised into "current" and "accumulating", relating to whether they reflect an instantaneous popularity or a general, time-independent popularity. The largest overlap in popularity rankings between the investigated data sources can be explained by the dimension of time dependence. It is present between the output of the page count predictor and the P2P rankings, the data sources behind both of which share an accumulating strategy of data storage. `Twitter` and `last.fm` on the other hand are more time dependent in that they reflect better the current "hotness" of an artist than his or her overall popularity.

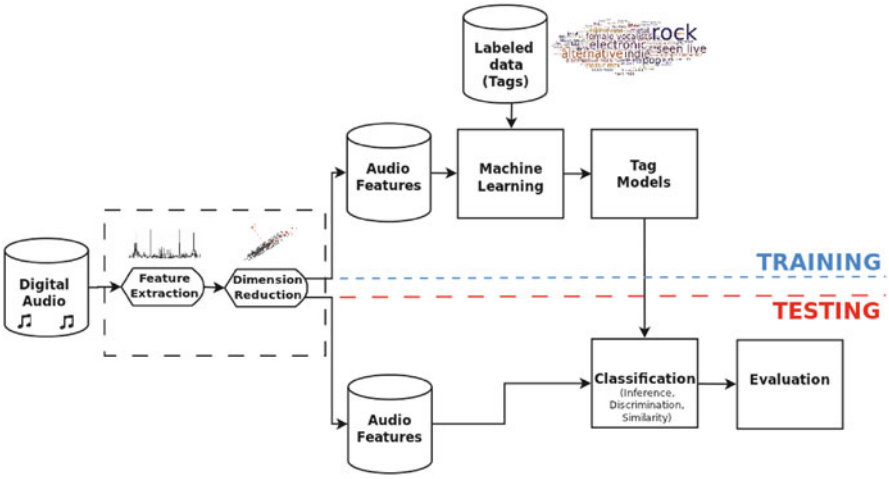

Fig. 6 Basic scheme of a music auto-tagger

3.3 Auto-tagging Music

Semantic labels attached to multimedia items, such as images, music pieces, or videos, have become an important means to categorise and describe such items and to communicate particular opinions or feelings about them by users of social media. The process of automatically attaching semantic labels, or tags, to music pieces is referred to as *auto-tagging* and is a rather recent research endeavour in MIR. Typically, first, a machine learning approach, a *supervised learner* to be more precise, is employed on a training data set that associates feature representations (commonly music content or music context features) with semantic tags. After training is finished, the classifier is used to predict labels to previously unseen music items. In order to increase computational efficiency, optionally some *feature selection* or *dimensionality reduction* technique might be employed to the input feature vectors before training the classifier. This is of particular importance when dealing with high-dimensional representations of music items, which are typically present when modelling music items via a multimodal approach, for instance, via a feature vector describing aspects of the music content and the music context [76]. It was shown by Sordo in [86] that a dimensionality reduction of 95% by applying *principal components analysis* (PCA) [44] to the `CAL500` data set [89] and 600-dimensional audio feature vectors does not significantly decrease accuracy but decreases computational costs considerably. Figure 6 depicts the general framework of an auto-tagger [86].

One can broadly categorise music-tagging efforts into approaches that learn relations between feature representations of music files and semantic tags, henceforth referred to as "computational approaches" and strategies to infer tags directly

Table 6 Comparing different approaches to tag music

Source/approach	Advantages	Disadvantages
Human surveys	Well-defined vocabulary, high-quality annotations, strong labelling	Restricted vocabulary, yields only small data sets, high human effort, time-consuming
Social tags	Unrestricted vocabulary, incorporates social and cultural context, wisdom of the crowds	Popularity bias, community bias, weak labelling
Games with a purpose	Wisdom of the crowds, entertainment factor yields high-quality and fast annotations	Cheating, tags valid only for short segments (incentive for quick skipping)
Web pages	Incorporates cultural context, large corpus available, no immediate human involvement necessary	Noisy annotations, weak labelling, sparseness in the "long tail"
Auto-tagging	Not affected by cold-start problem, no immediate human involvement necessary, strong labelling	Computationally expensive, limited by training data

from some kind of user input, referred to as "human-centred approaches", in the following. Both strategies will be addressed below. As for the former one, methods that learn tags from *co-occurrence data* (*collaborative filtering*), *audio features*, and *web pages* will be introduced. For the category of human-centred approaches, another *game with a purpose* will be presented, and the use of *music folksonomies* to infer tags and associate them to semantic categories will be discussed.

Turnbull et al. in [52] compare various data sources and corresponding algorithms (computational and human-centred) for the task of tagging music: *human surveys*, *social tags*, *games with a purpose*, *web pages*, and *auto-tagging*. They elaborate on advantages and disadvantages of each, which are summarised from [52] and extended by the author in Table 6. *Weak labelling* refers to the fact that one cannot infer from the absence of a tag t that t does not apply to the item. Users might simply not have thought of the tag in such a case. In contrast, a *strongly labelled* data set is complete in the sense that the absence of tag t does mean that t is not suited to describe the item under consideration. For an explanation of *popularity bias* and *community bias*, please refer to Sects. 3.1.2 and 3.2.1, respectively.

3.3.1 Computational Approaches

As described above, the most common approach to auto-tagging music is to train a classifier on music content features and learn relations between them and a set of

tags that are known to relate to the corresponding music items. As features typically rhythm and/or timbre descriptors are used [63], sometimes high-level features are included in addition [86].

Sordo proposes a simple and efficient algorithm based on a weighted vote k-nearest neighbour (kNN) classifier to propagate tags from training data to unseen music items [86]. Given a training set of PCA-compressed feature vectors of a song collection together with a set of labels for each piece, the proposed *weighted vote kNN* algorithm first determines the k closest neighbours to the seed song s, which should be tagged. The frequency of all tags assigned to s's neighbours are then summed up per tag, and a threshold relative to k ensures that only frequently used tags are predicted for s.

An approach similar to Sordo's is suggested in [46]. Kim et al. also employ a kNN classifier to address the problem of auto-tagging artists, but they analyse different artist similarity functions. They compare similarities derived from artist co-occurrences in `last.fm` playlists ("scrobbles"), from `last.fm` tags, from web pages about the artists, and from music content features. Using as ground truth tags manually assigned by music experts, Kim et al. found that the similarity measure based on `last.fm` co-occurrences performed best, both in terms of precision and recall. When using a kNN classifier, it is crucial to carefully select the similarity measure to determine the nearest neighbours. Depending on the origin of the features, a common choice is *cosine similarity* (for term-weighting features) or one of the distances/divergences *Mahalanobis*, *Manhattan*, or *Kullback–Leibler* (for music content features).

In their proposed algorithm to extract tags from artist-related web pages, Schedl et al. use a dictionary of musically relevant terms to filter the textual content of the pages under consideration [77]. The authors propose three different term-weighting functions to score the extracted tags per artist and predict the resulting top-ranked tags. A user survey was conducted to evaluate the quality of the suggested tags in terms of descriptiveness. Quite surprisingly, participants in the study found tags suggested by a simple document frequency function superior to those proposed by TF $\cdot$ IDF-based term scoring.

Mandel et al. [63] use *conditional restricted Boltzmann machines* [85] to learn tag language models over three sets of vocabularies: annotations by users of `Amazon`'s `Mechanical Turk`, of the tagging game `MajorMiner` [62], and of `last.fm`. The models are learned on the level of song segments. Optionally different "contexts" are included, that is, track level and user level annotations are factored in.

Seyerlehner et al. in [82] use a combination of different audio features described within their block-level framework [83]: *spectral pattern* (SP), *delta spectral pattern* (DSP), *variance delta spectral pattern* (VDSP), *logarithmic fluctuation pattern* (LFP), *correlation pattern* (CP), and *spectral contrast pattern* (SCP). Associations between songs and tags are then learned using a *random forest* classifier.

Recently, two-stage algorithms have become popular. In the first stage, they infer higher-level information from music content features, such as term weight vector representations. These new representations are then fed into a machine learning

Table 7 Most frequently used tags in the TagATune game, in descending order of frequency

classical, guitar, piano, violin, slow, strings, rock, techno, opera, drums, same, flute, fast, diff, electronic, ambient, beat, yes, harpsichord, indian, female, vocal, no, synth, quiet, no vocals, soft, sitar, no vocal, classic, male, singing, solo, vocals, cello, loud, woman, pop, male vocal, choir, violins, new age, beats, no voice, harp, voice, weird, instrumental, dance

algorithm to learn semantic labels [14, 65]. Sometimes this second stage is said to incorporate contextual aspects since correlations between tags are frequently considered. Alternatively, the term weight vectors inferred in the first stage can also be used as input to a music similarity measure [82]. As an example for a two-stage auto-tagger, Miotto et al. in [65] first model *semantic multinomials* over tags based on music content features. In order to account for co-occurrence relations between tags, they subsequently learn a *Dirichlet mixture model* of the semantic space, which eventually yields a *contextual multinomial*.

3.3.2 Human-Centred Approaches

Games with a purpose have already been introduced in Sect. 3.1.1, where it was shown how to use their results for similarity estimation. In [53], Law and von Ahn present another interesting game with a purpose that focuses on *input-agreement*. Users have to agree whether they are listening to the same piece of music or not, that is, they have to agree on the input. To this end, they can exchange any free-form text that helps to reach the goal. Usually players enter descriptive tags in an effort to quickly choose the correct one of the two classes "same" or "different". If they agree on the correct class, both players are awarded points and the next round starts. Each game lasts for a total of 3 min.

According to Law and von Ahn, this input-agreement mechanism offers the advantage of being more popular and producing a higher number of tags than other, similar games. Analysing the most frequently used tags (Table 7), most of them describe genre, instrumentation, and properties of the music. Due to the very nature of the game design, the top list also includes *communication tags* that are unsuited to describe the music itself ("yes", "same", "no", "diff"). Furthermore, *negation tags* are frequently used to indicate the absence of a particular musical aspect ("no vocal", for example).

Music *folksonomies* present another valuable source for musical information. They are created by large numbers of users via tagging particular music items with their own, specific vocabulary. Although this vocabulary is probably not as precise as the one employed by music experts, the wisdom of the crowds is potentially able to cover more diverse aspects of human music perception than experts can think of.

Sordo et al. [87] present a method to automatically categorise tags extracted from Wikipedia into semantically meaningful groups, which they call "facets".

Table 8 Top facets of music extracted from `Wikipedia`

Music genres
Music geography
Musical groups
Musicians
Musical culture
Occupations in music
Music people
Record labels
Music technology
Sociological genres of music

To this end, starting at the most generic page about "music", the authors extract links from `DBpedia`, a machine-readable knowledge base created from `Wikipedia` pages. Applying some heuristics, pages not related to music are filtered out. To the remaining nodes, the `PageRank` algorithm [67] is applied to determine a relevance score for all nodes/pages in the network. The top facets Sordo et al.'s method found on a data set of about 600,000 artists and 400,000 tags are given in Table 8. The facets and tags extracted in this way are particularly interesting for music retrieval systems, where the user might want to restrict the results to a search query to tracks that are similar according to a specific facet.

4 Conclusions and Research Directions

In this chapter, we have discussed how various kinds of social media can be used for common music information retrieval tasks. More precisely, approaches to infer *music similarity* from text and co-occurrence information were presented, strategies to estimate the *popularity* of a music item from social media were elaborated, and recent methods to automatically assign semantically meaningful tags to music items, a process also known as *auto-tagging*, were discussed.

I am sure that future research in music information retrieval has to strive for a holistic perspective in a sense that information is not only derived from the audio signal and from meta-data but from many different types of multimedia material. Given the spiralling success of social media, analysis and data mining of the respective sources will open unprecedented opportunities to elaborate truly personalised and context-aware music retrieval and recommendation systems. Some concrete challenges in the context of social media mining for music information retrieval are *analysing music video clips* (official music videos as well as user-generated versions) to infer descriptive information; *processing images* of album covers, of band photographs, and of concert snapshots taken by enthusiastic music aficionados; and *making sense of textual data* about music items (for instance, microblogs). In addition, *data fusion* techniques are required to build multifaceted models that describe both music items and listeners in order to eventually enable the next generation of intelligent music retrieval systems.

References

1. Aucouturier, J.-J., Pachet, F., Sandler, M.: "The way it sounds": timbre models for analysis and retrieval of music signals. IEEE Trans. Multimed. **7**(6), 1028–1035 (2005)
2. Baccigalupo, C., Plaza, E., Donaldson, J.: Uncovering affinity of artists to multiple genres from social behaviour data. In: Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR), Philadelphia (2008)
3. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval – The Concepts and Technology Behind Search,2nd edn. Pearson, Harlow (2011)
4. Baltrunas, L., Kaminskas, M., Ludwig, B., Moling, O., Ricci, F., Lüke, K.-H., Schwaiger, R.: InCarMusic: context-aware music recommendations in a car. In: International Conference on Electronic Commerce and Web Technologies (EC-Web), Toulouse (2011)
5. Baum, L.E., Petrie, T.: Statistical inference for probabilistic functions of finite state Markov chains. Ann. Math. Stat. **37**(6), 1554–1563 (1966)
6. Berenzweig, A., Logan, B., Ellis, D.P., Whitman, B.: A large-scale evaluation of acoustic and subjective music similarity measures. In: Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR), Baltimore (2003)
7. Blanco, R., Lioma, C.: Graph-based term weighting for information retrieval. Inf. Retr. **15**(1), 54–92 (2012)
8. Büttcher, S., Clarke, C.L.A., Cormack, G.V.: Information Retrieval: Implementing and Evaluating Search Engines. MIT, Cambridge (2010)
9. Casey, M.A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., Slaney, M.: Content-based music information retrieval: current directions and future challenges. Proc. IEEE **96**, 668–696 (2008)
10. Celma, O.: Music Recommendation and Discovery – The Long Tail, Long Fail, and Long Play in the Digital Music Space. Springer, Berlin (2010)
11. Celma, O., Lamere, P.: ISMIR 2007 tutorial: music recommendation. http://mtg.upf.edu/~ocelma/MusicRecommendationTutorial-ISMIR2007. Accessed Dec 2007. September 23–27 2007
12. Cohen, W.W., Fan, W.: Web-collaborative filtering: recommending music by crawling the web. WWW9/Comput. Netw. **33**(1–6), 685–698 (2000)
13. Cooley, J.W., Tukey, J.W.: An algorithm for the machine calculation of complex Fourier series. Math. Comput. **19**(90), 297–301 (1965)
14. Coviello, E., Chan, A.B., Lanckriet, G.: Time series models for semantic music annotation. IEEE Trans. Audio Speech Lang. Process. **19**(5), 1343–1359 (2011)
15. Cunningham, S.J., Downie, J.S., Bainbridge, D.: "The pain, the pain": modelling music information behavior and the songs we hate. In: Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR), London (2005)
16. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. **41**, 391–407 (1990)
17. Downie, J.S.: The scientific evaluation of music information retrieval systems: foundations and future. Comput. Music J. **28**, 12–23 (2004)
18. Ellis, D.P., Whitman, B., Berenzweig, A., Lawrence, S.: The quest for ground truth in musical artist similarity. In: Proceedings of 3rd International Conference on Music Information Retrieval (ISMIR), Paris (2002)
19. Geleijnse, G., Schedl, M., Knees, P.: The quest for ground truth in musical artist tagging in the social web era. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR), Vienna (2007)
20. Gouyon, F., Herrera, P., Gomez, E., Cano, P., Bonada, J., Loscos, A., Amatriain, X., Serra, X.: Content processing of music audio signals. In: Polotti, P., Roccheso, D. (eds.) Sound to Sense, Sense to Sound: A State-of-the-Art in Sound and Music Computing, pp. 83–160. Logos Verlag, Berlin GmbH (2008)

21. Grace, J., Gruhl, D., Haas, K., Nagarajan, M., Robson, C., Sahoo, N.: Artist ranking through analysis of on-line community comments. In: Proceedings of the 17th ACM International World Wide Web Conference (WWW 2008), Bejing (2008)
22. Grossman, D.A., Frieder, O.: Information Retrieval: Algorithms and Heuristics. Kluwer International Series on Information Retrieval. Springer, Dordrecht (2004)
23. http://developer.echonest.com/docs/method/get_hotttnesss. Accessed Jan 2010
24. http://developer.echonest.com/docs/method/get_top_hottt_artists. Accessed Jan 2010
25. http://echonest.com. Accessed Feb 2012
26. http://en.wikipedia.org/wiki/Billboard_Hot_100. Accessed May 2009
27. http://last.fm. Accessed Jan 2012
28. http://music.strands.com. Accessed Nov 2009
29. http://www.allmusic.com. Accessed Jan 2010
30. http://www.artofthemix.org. Accessed Feb 2008
31. http://www.bandmetrics.com. Accessed May 2010
32. http://www.bigchampagne.com. Accessed May 2010
33. http://www.exalead.com. Accessed Aug 2010
34. http://www.facebook.com. Accessed Feb 2012
35. http://www.freebase.com. Accessed Jan 2010
36. http://www.freedb.org. Accessed Feb 2008
37. http://www.google.com. Accessed Mar 2010
38. http://www.gracenote.com. Accessed Feb 2008
39. http://www.ip2location.com. Accessed Mar 2010
40. http://www.myspace.com. Accessed Nov 2009
41. http://www.spotify.com. Accessed Feb 2012
42. http://www.twitter.com. Accessed Jan 2012
43. http://www.youtube.com. Accessed Feb 2012
44. Jolliffe, I.T.: Principal Component Analysis. Springer, New York (1986)
45. Jones, K.S., Walker, S.S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments. Inf. Process. Manag. **36**, 779–808 (2000)
46. Kim, J.H., Tomasik, B., Turnbull, D.: Using artist similarity to propagate semantic information. In: Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR), Kobe (2009)
47. Knees, P., Pampalk, E., Widmer, G.: Artist classification with web-based data. In: Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR), Barcelona, pp. 517–524 (2004)
48. Knees, P., Schedl, M., Pohle, T., Widmer, G.: An innovative three-dimensional user interface for exploring music collections enriched with meta-information from the web. In: Proceedings of the 14th ACM International Conference on Multimedia, Santa Barbara (2006)
49. Koenigstein, N., Shavitt, Y.: Song ranking based on piracy in peer-to-peer networks. In: Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009), Kobe (2009)
50. Koenigstein, N., Shavitt, Y., Tankel, T.: 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 (KDD), Las Vegas (2008)
51. Kohonen, T.: Self-Organizing Maps. Springer Series in Information Sciences, vol. 30, 3rd edn. Springer, Berlin (2001)
52. Lanckriet, G., Turnbull, D., Barrington, L.: Five approaches to collecting tags for music. In: Proceedings of the 9th International Society for Music Information Retrieval Conference (ISMIR), Philadelphia (2008)
53. Law, E., von Ahn, L.: Input-agreement: a new mechanism for collecting data using human computation games. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI), Boston, pp. 1197–1206 (2009)
54. Law, E., von Ahn, L., Dannenberg, R., Crawford, M.: Tagatune: a game for music and sound annotation. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR), Vienna (2007)

55. Lee, J.H.: Analysis of user needs and information features in natural language queries seeking user information. J. Am. Soc. Inf. Sci. Technol. (JASIST) **61**, 1025–1045 (2010)
56. Levy, M., Sandler, M.: A semantic space for music derived from social tags. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR), Vienna (2007)
57. Li, T., Ogihara, M., Tzanetakis, G. (eds.): Music Data Mining. CRC/Chapman Hall, Boca Raton (2011)
58. Logan, B., Ellis, D.P., Berenzweig, A.: Toward evaluation techniques for music similarity. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR): Workshop on the Evaluation of Music Information Retrieval Systems, Toronto (2003)
59. Luhn, H.P.: A statistical approach to mechanized encoding and searching of literary information. IBM J. **1**, 309–317 (1957)
60. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Cam, L.M.L., Neyman, J. (eds.) Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability. Statistics, vol. I, pp. 281–297. University of California Press, Berkeley/Los Angeles (1967)
61. Mandel, M.I., Ellis, D.P.: A Web-based game for collecting music metadata. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR), Vienna (2007)
62. Mandel, M.I., Ellis, D.P.W.: A web-based game for collecting music metadata. J. New Music Res. **37**(2), 151–165 (2008)
63. Mandel, M.I., Pascanu, R., Eck, D., Bengio, Y., Aiello, L.M., Schifanella, R., Menczer, F.: Contextual tag inference. ACM Trans. Multimed. Comput. Commun. Appl. **7S**(1), 32:1–32:18 (2011)
64. McFee, B., Lanckriet, G.: The natural language of playlists. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR), Miami (2011)
65. Miotto, R., Barrington, L., Lanckriet, G.: Improving auto-tagging by modeling semantic co-occurrences. In: Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010), Utrecht (2010)
66. Pachet, F., Westerman, G., Laigre, D.: Musical data mining for electronic music distribution. In: Proceedings of the 1st International Conference on Web Delivering of Music (WEDELMUSIC), Florence (2001)
67. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: bringing order to the web. In: Proceedings of the Annual Meeting of the American Society for Information Science (ASIS), Pittsburgh, pp. 161–172 (1998)
68. Pohle, T., Knees, P., Schedl, M., Pampalk, E., Widmer, G.: "Reinventing the wheel": a novel approach to music player interfaces. IEEE Trans. Multimed. **9**, 567–575 (2007)
69. Robertson, S., Walker, S., Hancock-Beaulieu, M.: Large test collection experiments on an operational, interactive system: Okapi at TREC. Inf. Process. Manag. **31**, 345–360 (1995)
70. Robertson, S., Walker, S., Beaulieu, M.: Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive track. In: Proceedings of the 7th Text REtrieval Conference (TREC-7), Gaithersburg, pp. 253–264 (1999)
71. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
72. Schedl, M.: Automatically extracting, analyzing, and visualizing information on music artists from the world wide web. Ph.D. thesis, Johannes Kepler University Linz, Linz (2008)
73. Schedl, M.: On the use of microblogging posts for similarity estimation and artist labeling. In: Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR), Utrecht (2010)
74. Schedl, M.: #nowplaying madonna: a large-scale evaluation on estimating similarities between music artists and between movies from microblogs. Inf. Retr. **15**, 183–217 (2012)
75. Schedl, M., Knees, P.: Context-based music similarity estimation. In: Proceedings of the 3rd International Workshop on Learning the Semantics of Audio Signals (LSAS), Graz (2009)
76. Schedl, M., Knees, P.: Personalization in multimodal music retrieval. In: Proceedings of the 9th Workshop on Adaptive Multimedia Retrieval (AMR), Barcelona (2011)

77. Schedl, M., Pohle, T.: Enlightening the sun: a user interface to explore music artists via multimedia content. Multimed. Tools Appl. Spec. Issue Semant. Digit. Media Technol. **49**(1), 101–118 (2010)
78. Schedl, M., Knees, P., Widmer, G.: A web-based approach to assessing artist similarity using co-occurrences. In: Proceedings of the 4th International Workshop on Content-Based Multimedia Indexing (CBMI), Riga (2005)
79. Schedl, M., Pohle, T., Koenigstein, N., Knees, P.: What's hot? Estimating country-specific artist popularity. In: Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR), Utrecht (2010)
80. Schedl, M., Knees, P., Böck, S.: Investigating the similarity space of music artists on the micro-blogosphere. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR), Miami (2011)
81. Schedl, M., Hauger, D., Schnitzer, D.: A model for serendipitous music retrieval. In: Proceedings of the 16th International Conference on Intelligent User Interfaces (IUI 2012): 2nd International Workshop on Context-Awareness in Retrieval and Recommendation (CaRR 2012), Lisbon (2012)
82. Seyerlehner, K., Schedl, M., Knees, P., Sonnleitner, R.: A refined block-level feature set for classification, similarity and tag prediction. In: Extended Abstract to the Music Information Retrieval Evaluation eXchange (MIREX 2011)/12th International Society for Music Information Retrieval Conference (ISMIR 2011), Miami (2009)
83. Seyerlehner, K., Widmer, G., Pohle, T.: Fusing block-level features for music similarity estimation. In: Proceedings of the 13th International Conference on Digital Audio Effects (DAFx-10), Graz (2010)
84. Shavitt, Y., Weinsberg, U.: Songs clustering using peer-to-peer co-occurrences. In: Proceedings of the IEEE International Symposium on Multimedia (ISM): International Workshop on Advances in Music Information Research (AdMIRe), San Diego (2009)
85. Smolensky, P.: Information Processing in Dynamical Systems: Foundations of Harmony Theory, pp. 194–281. MIT, Cambridge (1986)
86. Sordo, M.: Semantic annotation of music collections: a computational approach. Ph.D. thesis, Universitat Pompeu Fabra, Barcelona (2012)
87. Sordo, M., Gouyon, F., Sarmento, L.: A method for obtaining semantic facets of music tags. In: Proceedings of the Workshop on Music Recommendation and Discovery (WOMRAD), Barcelona (2010)
88. Stober, S.: Adaptive methods for user-centered organization of music collections. Ph.D. thesis, Otto-von-Guericke-University, Magdeburg (2011)
89. Turnbull, D., Barrington, L., Torres, D., Lanckriet, G.: Towards musical query-by-semantic-description using the CAL500 data set. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Amsterdam (2007)
90. Turnbull, D., Liu, R., Barrington, L., Lanckriet, G.: A game-based approach for collecting semantic annotations of music. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR), Vienna (2007)
91. Vasconcelos, N.: Image indexing with mixture hierarchies. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Kauai (2001)
92. Whitman, B., Lawrence, S.: Inferring descriptions and similarity for music from community metadata. In: Proceedings of the 2002 International Computer Music Conference (ICMC), Göteborg, pp. 591–598 (2002)
93. Wolff, D., Weyde, T.: Adapting metrics for music similarity using comparative ratings. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR), Miami (2011)