Machine Learning Approaches for Music Information Retrieval

Chapter · January 2009

Source: InTech

CITATION
1

READS
82

4 authors, including:

Mitsunori Ogihara
University of Miami
259 PUBLICATIONS **7,775** CITATIONS
SEE PROFILE

Bo Shao
15 PUBLICATIONS **164** CITATIONS
SEE PROFILE

18

Machine Learning Approaches for Music Information Retrieval

Tao Li[1], Mitsunori Ogihara[2], Bo Shao[3] and DingdingWang[4]
[1]*School of Computer Science, Florida International University*
[2]*Department of Computer Science, University of Miami*
[3]*School of Computer Science, Florida International University*
[4]*School of Computer Science, Florida International University*
USA

1. Introduction

The rapid growth of the Internet and the advancements of Internet technologies have made it possible for music listeners to have access to a large amount of on-line music data, including music sound signals, lyrics, biographies, and discographies. Music artists in the 21st century are promoted through various kinds of websites that are managed by themselves, by their fans, or by their record companies. Also, they are subjects of discussions in Internet newsgroups and bulletin boards.

This raises the question of whether computer programs can enrich the experience of music listeners by enabling the listeners to have access to such a large volume of on-line music data. Multimedia conferences, e.g. ISMIR (International Conference on Music Information Retrieval) and WEDELMUSIC (Web Delivery of Music), have a focus on the development of computational techniques for analyzing, summarizing, indexing, and classifying music data. In [Huron, 2000] Huron points out that since the preeminent functions of music are social and psychological, the most useful characterizationwould be based on four types of information: *genre*, *emotion*, *style*, and *similarity*. The four types of characteristics are strongly related to each other. Certain emotional labels prominently apply to music in particular genres, e.g., "angry" for punk music, "depressed" for slow blues, and "happy" for children music. A style is often defined within a genre, e.g., "hard-bop jazz" and "American rock." Similar music pieces are likely to be those in the same genre, of the same style, and with the same emotional labeling. However, there are traits that distinguish them from the rest. Emotional labeling is transient, in the sense that the labels can be dependent on the state of mind of the listener, and popular music styles are perhaps defined not just in terms of sound signals but in terms of the way the lyrics are written, which is likely beyond the reach of sound feature extraction algorithms.

In this chapter, we briefly discuss various machine learning approaches used for recognizing the above four types of features music information retrieval. In particular, we investigate the following approaches: (1) multi-class classification for music genre categorization; (2) multi-label classification for emotion detection; (3) clustering for music style identification; and (4) semi-supervised learning for music recommendation. Parts of

Source: Theory and Novel Applications of Machine Learning, Book edited by: Meng Joo Er and Yi Zhou,
ISBN 978-3-902613-55-4, pp. 376, February 2009, I-Tech, Vienna, Austria

the work presented in this chapter have appeared in [Li & Zhu, 2006, Li & Ogihara, 2006, Li & Ogihara, 2003, Li et al., 2003, Li & Tzanetakis, 2003, Shao et al., 2008].
The rest of the chapter is organized as follows: Section 2 briefly introduces the feature extraction from music audio signals and lyrics; Section 3 discusses the multi-class classification methods for music genre categorization; Section 4 presents multi-label classification methods for emotion detection; Section 5 studies the bi-modal clustering for music style identification; Section 6 proposes a graph-based semi-supervised learning method for music recommendation; and Finally Section 7 concludes.

2. Music feature extraction

Before applying machine learning approaches in music information retrieval, an important step is the determination of the features extracted from music data. All the machine learning methods discussed in this chapter make use of the content features extracted from music audio signals. In addition, for music style identification, we also make use of the text-based features from music lyrics.

2.1 Content feature extraction

There has been a considerable amount of work in extracting descriptive features from music signals for music genre classification and artist identification [Foote & Uchihashi, 2001, Tzanetakis & Cook, 2002a, Logan & Salomon, 2001, Li et al., 2003]. In our study, we use timbral features along with wavelet coefficient histograms.

2.1.1 Mel-Frequency Cepstral Coefficients (MFCC)

Mel-Frequency Cepstral Coefficients (MFCC) is a feature set that is highly popular in speech processing. It is designed to capture short-term spectral-based features. The features are computed as follows: First, for each frame, the logarithm of the amplitude spectrum based on short-term Fourier transform is calculated, where the frequencies are divided into thirteen bins using the Mel-frequency scaling. Next, this vector is then decorrelated using discrete cosine transform. This is the MFCC vector. In this work, we use the first five bins, and compute the mean and variance of each over the frames.

2.1.2 Short-Term Fourier Transform Features (STFT)

This is a set of features related to timbral textures and is not captured using MFCC. It consists of the following five types: Spectral Centroid, Spectral Rolloff, Spectral Flux, Zero Crossings, and Low Energy. More detailed descriptions of STFT can be found in [Tzanetakis & Cook, 2002a].

2.1.3 DaubechiesWavelet Coefficient Histograms (DWCH)

Daubechies wavelet filters are a set of filters that are popular in image retrieval (For more details, see [Daubechies, 1992]). The DaubechiesWavelet Coefficient Histograms, proposed in [Li et al., 2003], are features extracted in the following manner: First, the Daubechies-8 (db_8) filter with seven levels of decomposition (or seven subbands) is applied to 30 seconds of monaural audio signals. Then, the histogram of the wavelet coefficients is computed at each subband. Then the first three moments of a histogram, i.e., the average, the variance, and the skewness, are calculated from each subband. In addition, the subband energy,

defined as the mean of the absolute value of the coefficients, is computed from each subband. More details of DWCH can be found in [Li et al., 2003].

2.2 Lyrics-based feature sets

To accommodate the characteristics of the lyrics, our text-based feature extraction consists of four components: bag-of-words features, Part-of-Speech statistics, lexical features and orthographic features.

- *Bag-of-words*: We compute the TF-IDF measure for each word and select top 200 words as our features. Stemming operations are not applied.
- *Part-of-Speech statistics*: We use the output of the part-of-speech (POS) tagger by Brill [Bill, 1994] as the basis for feature extraction. The POS statistics usually reflect the characteristics of writing. There are 36 POS features extracted from each document, one for each POS tag expressed as a percentage of the total number of words for the document.
- *Lexical Features*: By "lexical features" we mean the features of individual wordtokens in the text. The most basic lexical features are lists of 303 generic function words taken from [Mitton, 1987][1], which generally serve as proxies for choice in syntactic (e.g., preposition phrase modifiers vs. adjectives or adverbs), semantic (e.g., usage of passive voice indicated by auxiliary verbs), and pragmatic (e.g., first-person pronouns indicating personalization of a text) planes. Function words have been shown to be effective style markers.
- *Orthographic features*: We also use orthographic features of lexical items, such as capitalization, word placement, word length distribution as our features. Word orders and lengths are very useful since the writing of lyrics usually follows certain melody.

3. Music genre categorization

3.1 Problem overview

Here we study the problem of content-based music genre categorization, i.e., classification of music pieces into a single unique class based computational analysis of music feature representations. Automatic music genre classification is a fundamental component of music information retrieval systems. Once the content-based features have been extracted from music pieces, the problem of music genre categorization is reduced to a multi-class classification problem: identifying the genre labels for music pieces from a set of pre-defined genre categories based on the feature representation of music audio signals.

3.2 Method description

We test various classification algorithms for the actual classification: GMM (Gaussian Mixture Models) with three Gaussians, KNN (k-Nearest Neighbors) with k = 5, LDA (Linear Discriminant Analysis), and multi-class extensions of support vector machines (SVM). Support vector machines (SVM) [Vapnik, 1998] is a method that has shown superb performance in binary classification problems. Intuitively, it aims at searching for a hyperplane that separates the positive data points and the negative data points with maximum margin. The method was originally designed as a binary classification algorithm.

[1] See http://www.cse.unsw.edu.au/~min/ILLDATA/Function.word.htm.

Several binary decomposition techniques are known. We use one-againstthe-rest (denoted by S1) and pairwise (denoted by S2), which assemble judgments respectively of the classifiers for distinguishing one class from the rest and of the classifiers for distinguishing one class from another. We also use a multi-class objective function version of SVM, MPSVM [Fung & Mangasarian, 2001] (we use short-hand MPS to refer to this algorithm), which can directly deal with multi-class problems. For S1 and S2, our SVM implementation is based on the LIBSVM [Chang & Lin, 2001], a library for support vector classification and regression. For experiments involving SVM, we test linear, polynomial, and radius-based kernels. The results we show are the best of the three kernel functions.

K-Nearest Neighbors (KNN) is a non-parametric classifier. Theoretical results show that its error is asymptotically at most twice as large as the Bayesian error rate. KNN has been applied to various music sound analysis problems. Given K as a parameter, it finds the K nearest neighbors among training data and uses the categories of the K neighbors to determine the class of a given input. We use the parameter K to 5.

Gaussian Mixture Models (GMM) is a method that has been widely used in music information retrieval. The probability density function (pdf) for each class is assumed to consist of a mixture of a number of multidimensional Gaussian distributions. The iterative expectation-minimization (EM) algorithm is then used to estimate the parameters of each Gaussian component and the mixture weights.

Linear Discriminant Analysis (LDA) works by finding a linear transformation that best discriminates among classes. The classification is then performed in the transformed space using some metric such as Euclidean distances.

3.3 Experiments

We test the classification method on the dataset used in [Tzanetakis & Cook, 2002b], which consists of 1,000 30-second-long sound files covering ten genres with 100 files per genre. The ten genres are Blues, Classical, Country, Disco, Hiphop, Jazz, Metal, Pop, Reggae, and Rock. The files are collected from radio and CD's. The experimental results are presented in Figure 1. The average accuracy of the one-versus-the-rest classifiers over a ten-fold cross-validation test, as shown in Figure 2 is very high for all ten classes (ranging from 91 to 99 %).

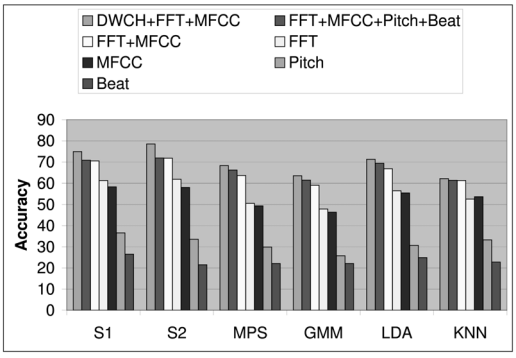

Fig. 1. The classification accuracy of the learning methods tested on the dataset using various combinations of features. The accuracy values are calculated via ten-fold cross validation.

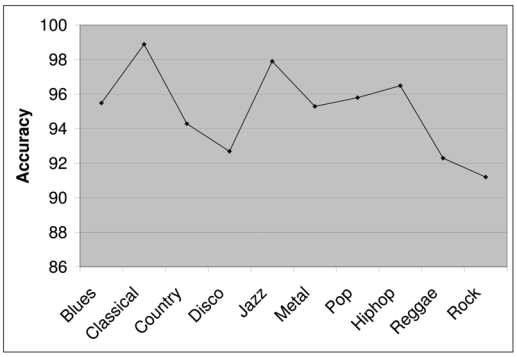

Fig. 2. The genre specific accuracy of the S1 method with DBCH, FFT and MFCC. The results are calculated via ten-fold cross validation.

Perrot and Gjerdingen [Perrot & Gjerdigen, 1999] report a human subject study in which college students were trained to learn a music company's genre classification on a ten-genre data collection, where the trained students achieved about 70% accuracy. Our results cannot be directly compared against their results because the datasets are different, but one can clearly say that the precision of the classification achieved here is satisfyingly high. The experiments demonstrate that music genre categorization can achieve good performances using machine learning techniques.

4. Emotion detection in music

4.1 Problem overview

Relations between musical sounds and their impact on the emotion of the listeners have been studied for decades. The celebrated paper of Hevner [Hevner, 1936] studied this relation through experiments in which the listeners are asked to write adjectives that came to their minds as the most descriptive of the music played. The experiments confirmed a hypothesis that music inherently carries emotional meaning. Hevner discovered the existence of clusters of descriptive adjectives and laid them out (there were eight of them) in a circle. She also discovered that the labeling is consistent within a group having a similar cultural background. The Hevner adjectives were refined and regrouped into ten adjective groups by Farnsworth [Farnsworth, 1958].

We cast the emotion detection problem as a *multi-label classification problem*, where the music sounds are classified into multiple classes simultaneously. That is a single music sound may be characterized by more than one label, e.g., both "dreamy" and "cheerful."

4.2 Method description

4.2.1 Multi-label classification

We resort to the scarcity of literature in multi-label classification by decomposing the problem into a set of binary classification problems. In this approach, for each binary problem a classifier is developed using the projection of the training data to the binary problem. To determine labels of a test data, the binary classifiers thus developed are run individually on the data and every label for which the output of the classifier exceeds a predetermined threshold is selected as a label of the data. See [Schapire & Singer, 2000] for

similar treatments in the text classification domain. To build classifiers we used Support Vector Machines [Vapnik, 1998].

4.2.2 The dataset and emotional labeling

A dataset consisting of 235 instrumental jazz tracks is used for the experiment. The dataset was constructed by the authors from the CD collection of the second author. The files are labeled independently by two subjects: a 39 year old male (subject 1) and a 25 year old male (subject 2). Each track is labeled using a scale ranging from −4 to +4 on each of three bipolar adjective pairs: (Cheerful versus Depressing), (Relaxing versus Exciting), and (Comforting versus Disturbing), where 0 is thought of as neutral. Our early work on emotion labeling [Li & Ogihara, 2003] uses binary labels (existence versus non-existence) based on the adjective groups of Farnsworth. The classification accuracy is not very high (around 60%). The low accuracy can be attributed to the presence of many labels to choose from. The recent experiments conducted by Leman *et al.* [Leman et al., 2005] using scales on ten bipolar adjective pairs suggest that variations in emotional labeling can be approximated using only spanned three major principal components, which are hard to name. With these results in mind we decided to generate three bipolar adjective pairs based on the eight adjective groups of Hevner.

4.3 Experiments

The accuracy of the performance is presented in Table 1. Here the accuracy measure is the Hamming accuracy, that is, the ratio of the number of True Positives and TrueNegative against the total number of inputs. In each measure, the tracks labeled 0 are altogether put on either the positive side or the negative side. It is clear that the accuracy of detection was always at least 70% and sometimes more than 80%. Also, there is a large gap in the performance between the two subjects on the first two measures. We observe that this difference is coming from the difference in the cultural background of the subjects. To deal with labeling of a much larger group of listeners one should cluster them into groups depending on their labeling and train the emotion detection system for each group.

Subject	Cheerful vs. Depressing	Relaxing vs. Exciting	Comforting vs. Disturbing
1	83.3 (8.0)	70.4 (9.9)	72.4 (5.1)
2	69.6 (10.0)	83.7 (7.3)	70.9 (9.1)

Table 1. The accuracy (in %) of emotion detection. Within parentheses are standard deviations.

5. Music style identification

5.1 Problem overview

This section addresses the issue of music style identification. Ellis *et al.* point out that similarity between artists reflects personal tastes and suggest that different measures have to be combined together so as to achieve reasonable results in similar artist discovery [Ellis et al., 2002]. We focus our attention to singer-song-writers, i.e., those who sing their own compositions. We take the standpoint that the artistic style of a singer-song-writer is reflected both in the acoustic sounds and in the lyrics. We therefore hypothesize that the

artistic styles of an artist can be captured better by combining acoustic features and linguistic features of songs than by using only one type of features. In this section, we describe our bi-modal clustering algorithms to group pop music pieces into groups with respect to the artists by using both acoustic features and linguistic features.

5.2 Method description

Our clustering algorithm is based on the basic principle of minimizing disagreement, i.e., minimizing the disagreement between two individual models could lead to the improvement of learning performance of individual models [Li & Ogihara, 2005]. The clustering algorithm is an extension of the EM method [Dempster et al., 1977]. In each iteration of algorithm, an EM type procedure is employed to bootstrap the model by starting with the cluster assignments obtained in the previous iteration. Upon convergence, the two individual models are used to construct the final cluster assignment. Table 2 lists the notions used for the algorithm and Figure 1 presents the algorithm procedure.

n	Number of Songs
$s_i = (s_i^1, s_i^2)$	A song s_i has two modes: content s_i^1 and lyrics s_i^2
$S = (s_1, \cdots, s_n)$	A collection of songs
K	Number of clusters
$\Lambda^1 = (\lambda_1^1, \cdots, \lambda_K^1)$	Modal 1 model parameters
$\Lambda^2 = (\lambda_1^2, \cdots, \lambda_K^2)$	Modal 2 model parameters
$Y = (y_1, ..., \cdots, y_n)$ $y_n \in \{1, \cdots, K\}$	Cluster assignment vector
$s \in S$	s represents a song from S
$y_s = k$	Song s is in k-th cluster

Table 2. The list of notations

We assume parameterized models, one for each cluster. Typically, all the models are from the same family, e.g., multivariate Gaussian. The algorithm described above is a variant of the EM algorithm. It performs an iterative optimization process for each data source by using the cluster assignments (possibly from another data source). Note that in each iteration, one data source is picked and every data point is reassigned to one of the clusters based on information from that data source and on its previous assignment. At the end of each iteration, the algorithm explicitly checks whether the agreement between two clusterings (one clustering from each data source) has been improved. If it is improved, the algorithm then continues to iterate. Otherwise, the algorithm will go back to the allocation step and hopefully get a new clustering.

5.3 Experiments

5.3.1 Data description

Our experiments are performed on the dataset consisting of 570 songs from 53 albums of a total of 41 artists. The sound recordings and the lyrics from them are obtained. To obtain the ground truth of song styles, we choose to use similarity information between artists available at All Music Guide artist pages (http://www.allmusic.com), assuming that this information is the reflection of multiple individual users. By examining All Music Guide artist pages, if the name of an artist X appears on the list of artists similar to Y, it is considered that X is similar to Y. The similarity graph of the 41 artists is shown in Figure 3.

Algorithm 1 : **Bimodal Clustering**

Input: S, K

Output: Cluster assignment Y as well as the trained model structure

1: **Initialization:** Initialize the model structure (Λ^1, Λ^2) as well as the cluster assignment Y

2: **while** the stopping criterion does not meet **do**

3: **Step I:**
Randomly pick a different data source $i \in \{1, 2\}$

4: **Step II:**
Model Re-estimation for source i: for each cluster k, the model parameters, λ_k^i, are re-estimated as

$$\lambda_k^i = \underset{\lambda}{\operatorname{argmax}} \sum_{s:s\in S, y_s=k} \log P(s^i|\Lambda^i)$$

5: **Step III:**
Sample re-assignment: for each data sample $s \in S$, set

$$y_s = \underset{k}{\operatorname{argmax}} \log P(s^i|\lambda_k^i)$$

6: **Step IV:**
Measure the agreement between two sources. If the agreement increases, goto Step I. Otherwise, goto Step II.

7: **end while**

8: Return Y as well as the trained models (Λ^1, Λ^2)

We select artists having a large number of neighbors. There are three of them, Fleetwood Mac, Yes, and Utopia. These three are neighbors to one another, so we select the neighbors of these three as a cluster. Of the remaining nodes we identify two other clusters in a similar

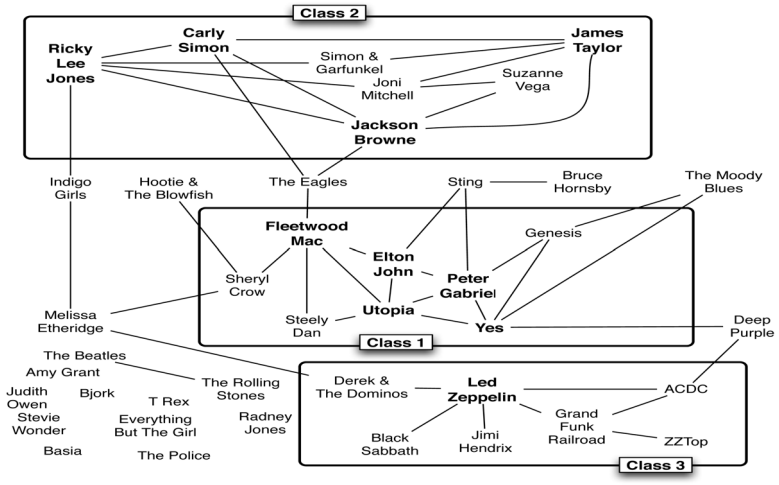

Fig. 3. The artist similarity graph. The names in bold are "core" nodes.

manner. All the remaining artists are in a separated cluster. The clusters are listed in Table 3. Our goal is to identify the song styles using both content and lyrics, i.e., cluster the 570 songs into the four different clusters. We use the cluster information of the artists as the labels for their songs.

Clusters	Members
No. 1	{ *Fleetwood Mac, Yes, Utopia, Elton John, Genesis, Steely Dan, Peter Gabriel* }
No. 2	{ *Carly Simon, Joni Mitchell, James Taylor, Suzanne Vega, Ricky Lee Jones, Simon & Garfunkel* }
No. 3	{ *AC/DC, Black Sabbath, ZZ Top, Led Zeppelin, Grand Funk Railroad, Derek & The Dominos* }
No. 4	All the remaining artists

Table 3. Cluster Memberships.

We use Purity and Accuracy [Zhao and Karypis, 2004, Ding et al., 2006] as our performance measures. Purity measures the extent to which each cluster contains data points from primarily one class [Zhao and Karypis, 2004]. In general, the larger the values of purity, the better the clustering solution is. Accuracy discovers the one-toone relationship between clusters and classes, therefore measure the extent to which each cluster contains data points from the corresponding class [Ding et al., 2006]. It sums up the whole matching degree between all pair class-clusters. The larger accuracy usually means the better clustering performance.

5.3.2 Result analysis

We compare the results of the bimodal clustering algorithm with the results obtained when the clustering is applied on the two sources of data separately. We also compare the bimodal clustering algorithm with the following clustering strategies on integrating different information sources: (1) Feature-Level Integration: Feature-level integration performs K-means clustering after simply concatenating the features obtained from the two data sources. (2) Cluster Integration: Cluster integration refers to the procedure of obtaining a combined clustering from multiple clusterings of a dataset [Strehl & Ghosh, 2003, Monti et al., 2003, Gionis et al., 2005]. Formally, let $C_1^1, ..., C_1^{k_1}$denote the clusters obtained from source 1, and $C_2^1, ..., C_2^{k_2}$denote the clusters obtained from source 2. Each point d_i can be represented as a ($k_1 + k_2$)- dimensional vector

$$d_i = (d_{i11}, \cdots, d_{i1k_1}, \cdots, d_{i21}, \cdots, d_{i2k_2})$$

$$d_{ijl} = \begin{cases} 1 & d_i \in C_j^{k_j} \\ 0 & \text{otherwise} \end{cases}, \text{ for } 1 \leq j \leq 2.$$

A combined clustering can be found by applying the K-means algorithm on the new representation. (3) Sequential Integration: Sequential integration is an intermediate approach of combining different information sources. It first performs clustering on one

data source and obtains a clustering assignment, say, $C^1, ..., C^{k_1}$. We can represent each point d_i as a k_1-dimensional vector using the similar idea in cluster integration. Then we can combine the new representation with another data source using feature integration. Clustering can thus be performed on the new concatenated vectors. Depending on the order of the two sources, we have two sequential integration strategies:

a. Sequential Integration I: firstly cluster based on content, then integrate with lyrics;
b. Sequential Integration II: firstly cluster based on lyrics, then integrate with content.

We compare the results of bimodal clustering with the results obtained when clustering is applied on content and lyrics separately, and with the results of other integration strategies. Table 4 presents the experimental results. From the table, we observe the following: (1) The performance of purity and accuracy relative to the other is not always consistent in our comparison, i.e., higher purity values do not necessarily correspond to higher accuracy values. This is because different evaluation measures consider different aspects of the clustering results. (2) The purity and accuracy of feature-level integration are worse than those of content-only and lyric-only clustering methods. This shows that even though the joint feature space is often more informative than that available from individual sources, naive feature integration tends to generalize poorly [Wu et al., 1999]. (3) Cluster Integration: The cluster integration performs better than content-only and lyrics-only: cluster integration has higher purity and accuracy values than those of content-only and lyrics-only. This actually conforms to the results in [Gionis et al., 2005]: cluster aggregation would usually provide better clustering results. (4) Sequential Integration: the results of sequential integration are generally better than feature-level integration, and they are comparable with those of content-only and lyrics-only. (5) Our bimodal clustering outperforms all other methods in all three performance measures. The bimodal clustering algorithm can be thought as a kind of *semantic* integration of data from different information sources. The performance improvements show that bimodal clustering has advantages over cluster integration. The bimodal clustering aims to minimize the disagreements between different sources and it can implicitly learn the correlation structure between different sets of features.

Feature Set(s)	Purity	Accuracy
Content-only	0.436	0.438
Lyrics-only	0.444	0.402
Feature-Level Integration	0.425	0.380
Cluster Integration	0.465	0.423
Sequential Integration I	0.431	0.434
Sequential Integration II	0.438	0.407
bimodal Clustering	**0.471**	**0.453**

Table 4. Performance Comparison. The numbers are obtained by averaging over ten trials.

Experimental comparisons show that our bimodal clustering can efficiently identify song styles. For example, in our experiments, two songs from the album *Utopia / Anthology*: *Overture Mountain Top And Sunrise Communion With The Sun* and *The Very Last Time* would be put into two different clusters based on their contents or lyrics only. However, using both the content and lyrics, our bimodal clustering algorithm identifies them to be in the same cluster with similar styles. Similarly, bimodal clustering identifies two songs from the album *Peter-Gabriel / Peter Gabriel*: *Excuse Me* and *Solsbury hill* to be in the same cluster while other

methods do not. In our experiments, we have identified around 50 such pairs and they give good anecdotal evidence that our bi-modal clustering algorithm can efficiently identify song styles.

6. Music recommendation

6.1 Problem overview

Music recommendation is the problem of delivering to a music listener a list of music pieces that he/she is likely to enjoy listening to. Music recommendation has been receiving a growing amount of attention recently [Uitdenbogerd & van Schyndel, 2002, Oliver & Kreger-Stickles, 2006, Pauws et al., 2006, Platt et al., 2002, Cai et al., 2007, Logan, 2004, Chen & Chen, 2001]. Our goal for music recommendation is to satisfy the following two requirements:

- *High recommendation accuracy.* A good recommendation system should output a relatively short list of songs in which many pieces are favored and few pieces are not favored.
- *High recommendation novelty.* Good novelty is defined as rich artist variety and well-balanced music content variety. Music content represents the information of genre, timbre, pitch, and rhythm, and so on [Tzanetakis & Cook, 2002a]. Well-balance means that the music content needs to be diversified and informative while not diverging much from the user preferences.

Various approaches for making music recommendations have been developed by utilizing the demographic information of the listeners, the contents of the music, the user listening history, and the discography (e.g., Last.fm, Goombah, and Pandora). These approaches can be generally divided into two types: collaborative-filtering methods and content-based methods. *Collaborative-filtering methods* recommend songs by identifying similar users or items based on ratings of items given by users [Cohen & Fan, 2000, Breese et al., 1998, Herlocker et al., 1999]. If the rating of an item by a user is unavailable, collaborative-filtering methods estimate it by computing a weighted average of known ratings of the item by similar users. *Content-based methods* provide recommendations based on the features extracted from audio signals of songs [Huang & Jenor, 2004, Knees et al., 2006, Li et al., 2004, Li a& Ogihara, 2004] and/or on the meta-data including genre, styles, artists, and lyrics [Pauws et al., 2006, Ragno et al., 2005, Yoshii et al., 2006]. That the contents are susceptible to feature extraction makes music recommendation different from movie recommendation, in which the meta-data is generally the only source of information available [Melville et al., 2002]. Recent proposals of probabilistic models and hybrid algorithms [Popescul et al., 2001, Jung et al., 2004, Yoshii et al., 2006] are designed by combining contents and user ratings [Popescul et al., 2001, Jung et al., 2004, Yoshii et al., 2006].

6.2 Method description

Here we introduce a strategy for music recommendation by way of mixing collaborative filtering and acoustic contents of music. The method proposed here uses a novel dynamic music similarity measurement that utilizes the access patterns from large numbers of users and uses an undirected graph as representation. Recommendation is calculated using the graph Laplacian and label propagation defined over the graph.

6.2.1 Dynamic music similarity measurement

The music features are vectors in a multi-dimensional space, and the distance between the representation vectors characterizes and quantifies the closeness between two pieces of music. Traditionally there are two popular distance functions for measuring similarity in multimedia retrieval [Foote et al., 2002, Logan & Salomon, 2001, Rui & Huang, 2000]: *Minkowski Distance Function* and *Weighted Minkowski Distance Function*. In the Minkowski distance measurement, every audio feature is assigned with the equal weight when determining the similarity of music. This could be inappropriate given that people are more sensitive to certain acoustic features than the others. In addition, it is well known that the perception of music is subjective to individual users. Different users can have totally different opinions for the same pieces of music. Using a fixed set of weights for acoustic features is likely to fail in accounting for the taste of individual users. It is thus important to design an automatic scheme for assigning weights to audio features based on the taste of individual users.

We propose a novel dynamic similarity measurement that utilizes the access patterns of music from a considerable number of users. Our measurement scheme is based on the assumption that two pieces of music are similar in human perception when they share similar access patterns across multiple users. Here we cast the problem of computing the appropriate similarity measures as a learning problem whose goal is to assign approximate weights to each feature [Wettschereck & Aha, 1995]. To determine automatically the weights for audio features, we explore the metric learning approach [He et al., 2004, Xing et al., 2003], which learns appropriate similarity metrics based on the correlation between acoustic features and user access patterns of music.

6.2.2 Label propagation on graph

Once we compute the similarities between pairs of songs, we can then construct the song graph. Once we obtain the song graph, music recommendation can be viewed as label propagation from labeled data (i.e., items with ratings) to unlabeled data.

In its simplest form, the label propagation is like a random walk on a song graph $\mathcal{G}$ [Szummer & Jaakkola, 2001, Kondor & Lafferty, 2002, Smola & Kondor, 2003, Zhu et al., 2003, Zhou et al., 2003]. Here we use the Green's function of the Laplace operator for music recommendation [Ding et al., 2007].

Given a graph with edge weights W, the *combinatorial Laplacian* is defined to be $L = D - W$, where D is the diagonal matrix consisting of the row sums of W; i.e., $D = \mathrm{diag}(W\mathbf{e})$, $\mathbf{e} = (1 \cdots 1)^T$.

Green's function is defined on the generalized eigenvectors of the Laplacian matrix:

$$L\mathbf{u}_k = \zeta_k D\mathbf{u}_k, \quad \mathbf{u}_p^T D\mathbf{u}_q = \mathbf{z}_p^T \mathbf{z}_q = \delta_{pq}. \tag{1}$$

where $0 = \zeta_1 \leq \zeta_2 \leq \cdots \leq \zeta_n$are the eigenvalues and the zero-mode is the first eigenvector $\mathbf{u}_1 = \mathbf{e}/\sqrt{n}$. Then we have

$$G = \frac{1}{(D-W)_+} = \sum_{k=2}^{n} \frac{\mathbf{u}_k \mathbf{u}_k^T}{\zeta_k}. \tag{2}$$

In practice, we truncate the expansion after some K terms and store the K vectors. Green's function is computed on the fly. So the storage requirement is $O(Kn)$.

Let $\mathbf{y}^T = (y_1, \ldots, y_n)$ be the rating for a user. Suppose we are Given an incomplete Rating $\mathbf{y}_0^T$ = (5, ?, ?, 4, 2, ?, ?, ?, 3) and the song graph illustrated in Figure 4.

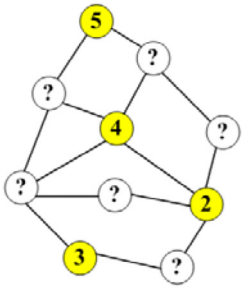

Fig. 4. An illustration of a recommendation task. The colored (shaded) nodes represent the rated items with their corresponding ratings. The others are the unrated items, whose ratings are unknown.

Our goal is to predict those missing values. Using Green's function, we initialize $\mathbf{y}_0^T$ = (5, 0, 0, 4, 2, 0, 0, 0, 3), and then compute the complete rating as the linear influence propagation

$$\mathbf{y} = G\mathbf{y}_0, \tag{3}$$

where G is the Green function built from the song graph.

6.2.3 Music ranking

After label propagation, we obtain the ratings for unrated songs and many of them might have the same rating. In practice, we might need a ranked list of the items to be recommended. The music ranking over a song graph $\mathcal{G}$ can be treated as the problem of finding the shortest path from the seed song node to the rest of the nodes in the song graph. The edges with low similarity have already been eliminated, so only the remaining edges can be used to construct shortest paths.

6.3 Experiments

The music data come from http://www.newwisdom.net. It is an educational and entertainment website in Chinese. It has about 4000 registered users visiting its forums regularly. They also listen songs and create playlists (called CDs on this website) as they prefer. Currently the website has a collection of more than 8,000 songs and hundreds of playlists.

By combining the user access pattern data with the content features of the songs, we generate the weight for each feature using the dynamic weighting scheme described above. Then we use the music ranking algorithm aforementioned to output the desired number of music pieces as our recommendations. In the experiments, the values of the ratings for the seed songs are set to be the same.

6.3.1 Comparison of different recommendation approaches

To demonstrate the performance of our music recommendation system, we compare the performance of the following five approaches:

1. **Content-based Approach (CBA)** This is solely based on acoustic content features extracted from the pieces of songs.
2. **Artist-based Approach (ABA)** This is solely based on artist, namely, it recommends songs only from the same artist.
3. **Access-pattern-based Approach (APA)** This is based on user access patterns. It selects the top songs with the highest co-occurrence frequency in the same playlists with the input song. This can also be thought as the item-based collaborative filtering method.
4. **Hybrid Approach (HA)** This is the approach explained in Section 1. It tries to integrate the collaborative filtering method and content-based method based on the algorithms described in [Jung et al., 2004].
5. **DWA** This is based on our approach, which first utilizes user access patterns to dynamically learn weights for each content features and then perform label propagation and ranking for music recommendation.

We conduct several sets of experiments to compare the performance of the listed approaches. The first comparison is designed to test the recommendation novelty and the playlist generation experiment is to examine the recommendation prediction ability, while the user study conducted is to assess the overall recommendation performance from the viewpoints of the end users.

6.3.2 Comparison on content variety

In this experiment, we evaluate if content variety as described in 1 are well balanced in different approaches. First of all, we cluster the 2829 songs using K-means algorithm according to their content features, and then, we study how many clusters the 10 songs recommended by each approach belong to. Also, we calculate the average distance among the 10 recommended songs of each of the 2829 seed songs using their content features. The more the clusters and/or the larger the distances, the more diverse the 10 songs, i.e. the more opportunity to get novel recommendation results.

Approach	Mean of Average Distance	Mean of Average Number of Clusters
CBA	2.55	2
ABA	8.74	5
APA	10.01	6
HA	5.28	4
DWA	5.88	4

Table 5. Results for content variety comparison.

From the experimental results listed in Table 5, we clearly observe that contentbased approach recommends songs with the highest content similarity, and the variety is very low. On the contrary, the access-pattern-based approach and the artist-based approach are diverse enough but lack of content similarity. Hybrid approach and our dynamic-weighting approach have comparable performance in well-balancing the content variety.

6.3.3 Comparison on playlist generation

Since playlists are generally a good means to reflect the interests of users, by comparing how accurate we can generate the whole original playlists from part of songs in them using

different methods, we can analyze the ability of the approaches to predict the interests and preferences of the users.

In this set of experiments, we randomly select 200 playlists from the dataset of 274 playlists, and run hybrid approach and our dynamic-weighting approach on the data for the two approaches to learn. Then we randomly select 5 songs from each of the rest 74 playlists, and generate 74 new playlists, each of which contains 50 distinct songs based on the ordered recommendation lists of the these 5 songs. Then we check how many of the songs in the rest of each original playlists (the number of songs available for checking varies from 5 to 15) match the songs in the new larger playlists. Figure 5 lists the boxplot results of the comparison among content-based approach, hybrid approach, and our dynamic-weighting approach. From Figure 5 and Table 6, we observe that our approach outperforms content-based approach and the hybrid approach.

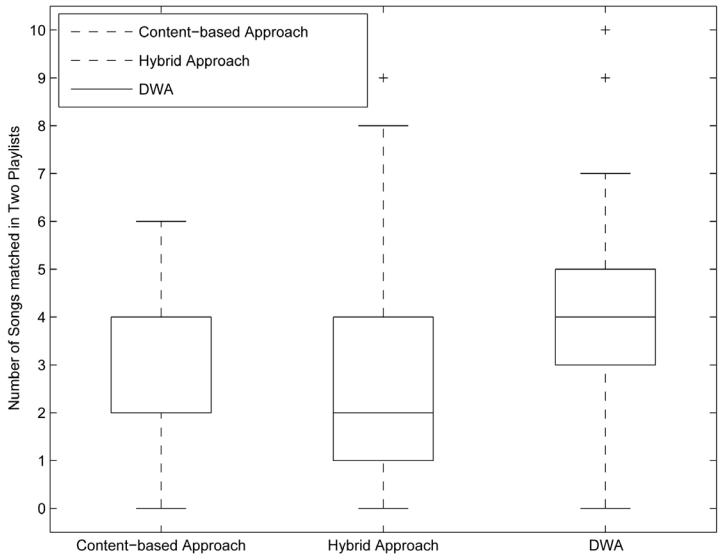

Fig. 5. Number of songs matched in user playlists and the playlists generated by different approaches

Approach	CBA	HA	DWA
Winning Rounds	8	20	37

Table 6. Times of one approach outperforms the other two by comparing the matches in two playlists

7. Conclusions

We discussed the following machine learning approaches used in music information retrieval: (1) multi-class classification methods for music genre categorization; (2) multi-label classification methods for emotion detection; (3) clustering methods for music style identification; and (4) semi-supervised learning methods for music recommendation. Experimental results are also presented to evaluate the approaches.

8. References

[Bill, 1994] Bill, E. (1994). Some advances in transformation-based parts of speech tagging. In *Proceedings of the twelfth national conference on Artificial intelligence (vol. 1)*, pages 722–727. American Association for Artificial Intelligence.

[Breese et al., 1998] Breese, J. S., Heckerman, D., and Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence*, pages 43–52.

[Cai et al., 2007] Cai, R., Zhang, C., Zhang, L., and Ma,W.-Y. (2007). Scalable music recommendation by search. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 1065–1074.

[Chang and Lin, 2001] Chang, C.-C. and Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm.

[Chen and Chen, 2001] Chen, H.-C. and Chen, A. L. P. (2001). A music recommendation system based on music data grouping and user interests. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 231–238, New York, NY, USA. ACM.

[Cohen and Fan, 2000] Cohen,W.W. and Fan,W. (2000). Web-collaborative filtering: recommending music by crawling the web. *Comput. Networks*, 33(1-6):685–698.

[Daubechies, 1992] Daubechies, I. (1992). *Ten lectures on wavelets*. SIAM, Philadelphia.

[Dempster et al., 1977] Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1.

[Ding et al., 2007] Ding, C., Jin, R., Li, T., and Simon, H. D. (2007). A learning framework using green's function and kernel regularization with application to recommender system. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 260–269, New York, NY, USA. ACM.

[Ding et al., 2006] Ding, C., Li, T., Peng, W., and Park, H. (2006). Orthogonal nonnegative matrix t-factorizations for clustering. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 126–135, New York, NY, USA. ACM.

[Ellis et al., 2002] Ellis, D., Whitman, B., Berenzweig, A., and Lawrence, S. (2002). The quest for ground truth in musical artist similarity. In *Proceedings of 3rd International Conference on Music Information Retrieval*, pages 170–177.

[Farnsworth, 1958] Farnsworth, P. R. (1958). *The social psychology of music*. The Dryden Press.

[Foote et al., 2002] Foote, J., Cooper, M., and Nam, U. (2002). Audio retrieval by rhythmic similarity. pages 265–266.

[Foote and Uchihashi, 2001] Foote, J. and Uchihashi, S. (2001). The beat spectrum: a new approach to rhythm analysis. In *IEEE International Conference on Multimedia & Expo 2001*.

[Fung and Mangasarian, 2001] Fung, G. and Mangasarian, O. L. (2001). Multicategory proximal support vector machine classifiers. Technical Report 01-06, University of Wisconsin at Madison.

[Gionis et al., 2005] Gionis, A., Mannila, H., and Tsaparas, P. (2005). Clustering aggregation. In *ICDE*, pages 341–352.

[He et al., 2004] He, X., Ma, W.-Y., and Zhang, H.-J. (2004). Learning an image manifold for retrieval. In *Proceedings of ACM MM 2004*.

[Herlocker et al., 1999] Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *SIGIR'99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237.

[Hevner, 1936] Hevner, K. (1936). Experimental studies of the elements of expression in music. *American Journal of Psychology*, 48:246–268.

[Huang and Jenor, 2004] Huang, Y.-C. and Jenor, S.-K. (2004). An audio recommendation system based on audio signature description scheme in mpeg-7 audio. In *2004 IEEE International Conference on Multimedia and Expo*, volume 1, pages 639–642.

[Huron, 2000] Huron, D. (2000). Perceptual and cognitive applications in music information retrieval. In *Proceedings of International Symposium on Music Information Retrieval*.

[Jung et al., 2004] Jung, K.-Y., Park, D.-H., and Lee, J.-H. (2004). Hybrid collaborative filtering and content-based filtering for improved recommender system. In *Computational Science - ICCS 2004*, pages 295–302. Springer Berlin / Heidelberg.

[Knees et al., 2006] P. Knees, T. Pohle, M. Schedl, and G.Widmer. Combining audiobased similarity with web-based data to accelerate automatic music playlist generation. In *Proceedings of the Seventh ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 147–154, ACM Press, 2006.

[Kondor and Lafferty, 2002] Kondor, R. and Lafferty, J. (2002). Diffusion kernels on graphs and other discrete input spaces. In *Proceedings of the 2002 International Conference on Machine Learning (ICML)*.

[Li et al., 2004] Li, Q., Kim, B.-M., Guan, D.-H., and Oh, D.-W. (2004). A music recommender based on audio features. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 532–533, New York, NY, USA. ACM.

[Li and Ogihara, 2003] Li, T. and Ogihara, M. (2003). Detecting emotion in music. In *Proceedings of the Fifth International Symposium on Music Information Retrieval (ISMIR2003)*, pages 239–240.

[Li and Ogihara, 2004] Li, T. and Ogihara, M. (2004). Content-based music similarity search and emotion detection. In *Proceedings of 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 705–708.

[Li and Ogihara, 2005] Li, T. and Ogihara, M. (2005). Semi-supervised learning from different information sources. *Knowledge and Information Systems Journal*, 7(3):289–309.

[Li and Ogihara, 2006] Li, T. and Ogihara, M. (2006). Toward intelligent music information retrieval. *IEEE Transactions on Multimedia*, 8(3):564–574.

[Li et al., 2003] Li, T., Ogihara, M., and Li, Q. (2003). A comparative study on content-based music genre classification. In *Proceedings of SIGIR*, pages 282–289.

[Li and Tzanetakis, 2003] Li, T. and Tzanetakis, G. (2003). Factors in automatic musical genre classification of audio signals. In *Proceedings of 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'03)*, pages 143–146. IEEE Computer Society.

[Li and Zhu, 2006] Li, T. and Zhu, M. O. S. (2006). Integrating features from different sources for music information retrieval. In *Proceedings of the Sixth IEEE International Conference on Data Mining (ICDM'06)*, pages 372–381.

[Logan, 2004] Logan, B. (2004). Music recommendation from song sets. In *Proceedings of ISMIR 2004*, pages 425–428.

[Logan and Salomon, 2001] Logan, B. and Salomon, A. (2001). A content-based music similarity function. Technical Report CRL 2001/02, Cambrige Research Laboratory.

[Melville et al., 2002] Melville, P., Mooney, R., and Nagarajan, R. (2002). Contentboosted collaborative filtering for improved recommendations. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI-02)*.

[Mitton, 1987] Mitton, R. (1987). Spelling checkers, spelling correctors and the misspellings of poor spellers. *Information Processing and Management*, 23(5):103–209.

[Leman et al., 2005] M. Leman, V. Vermeulen, L. De Voogdt, D. Moelants, and M. Lesaffre. Prediction of musical affect using a combination of acoustic structural cues. Journal of New Music Research, 34(1):39–67, 2005.

[Monti et al., 2003] Monti, S., Tamayo, P., Mesirov, J., and Gloub, T. (2003). Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning Journal*, 52(1-2):91–118.

[Oliver and Kreger-Stickles, 2006] Oliver, N. and Kreger-Stickles, L. (2006). Papa: Physiology and purpose-aware automatic playlist generation. In *Proceedings of the 7th International Conference on Music Information Retrieval*, pages 250–253.

[Pauws et al., 2006] Pauws, S., Verhaegh,W., and Vossen, M. (2006). Fast generation of optimal music playlists using local search. In *Proceedings of the 7th International Conference on Music Information Retrieval*, pages 138–143.

[Perrot and Gjerdigen, 1999] Perrot, D. and Gjerdigen, R. R. (1999). Scanning the dial: an exploration of factors in the identification of musical style. In *Proceedings of the 1999 Society for Music Perception and Cognition*, page 88.

[Platt et al., 2002] Platt, J. C., Burges, C. J. C., Swenson, S., Weare, C., and Zheng, A. (2002). Learning a gaussian process prior for automatically generating music playlists. In *Advances in Neural Information Processing Systems 14*, pages 1425–1432.

[Popescul et al., 2001] Popescul, A., Ungar, L., Pennock, D., and Lawrence, S. (2001). Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *17th Conference on Uncertainty in Artificial Intelligence*, pages 437–444, Seattle, Washington.

[Ragno et al., 2005] Ragno, R., Burges, C. J. C., and Herley, C. (2005). Inferring similarity between music objects with application to playlist generation. In *MIR'05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 73–80, New York, NY, USA. ACM.

[Rui and Huang, 2000] Rui, Y. and Huang, T. S. (2000). Optimizing learning in image retrieval. In *Proceedings of IEEE Computer Vision and Pattern Recognition*, pages 236–243.

[Schapire and Singer, 2000] Schapire, R. E. and Singer, Y. (2000). Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168.

[Shao et al., 2008] Shao, B., Wang, D., Li, T., and Ogihara., M. (2008). Music recommendation based on acoustic features and user access patterns. Manuscript in submission.

[Smola and Kondor, 2003] Smola, A. J. and Kondor, R. (2003). Kernels and regularization on graphs. In *Proceedings of the 16th Annual Conference on Learning Theory and 7th Kernel Workshop*, pages 144–158.

[Strehl and Ghosh, 2003] Strehl, A. and Ghosh, J. (2003). Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617.

[Szummer and Jaakkola, 2001] Szummer, M. and Jaakkola, T. (2001). Partially labeled classification with markov random walks. In *Advances in Neural Information Processing Systems*, volume 14.

[Tzanetakis and Cook, 2002a] Tzanetakis, G. and Cook, P. (2002a). Music genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*,10(5):293–302.

[Tzanetakis and Cook, 2002b] Tzanetakis, G. and Cook, P. (2002b). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5).

[Uitdenbogerd and van Schyndel, 2002] Uitdenbogerd, A. and van Schyndel, R. (2002). A review of factors affecting music recommender success. In *Proceedings of ISMIR*.

[Vapnik, 1998] Vapnik, V. N. (1998). *Statistical learning theory*. John Wiley & Sons, New York.

[Wettschereck and Aha, 1995] Wettschereck, D. and Aha, D. W. (1995). Weighting features. In Veloso, M. and Aamodt, A., editors, *Case-Based Reasoning, Research and Development, First International Conference*, pages 347–358. Springer-Verlag, Berlin.

[Wu et al., 1999] Wu, L., Oviatt, S. L., and Cohen, P. R. (1999). Multimodal integration - a statistical view. *IEEE Transactions on Multimedia*, 1(4):334–341.

[Xing et al., 2003] Xing, E. P., Ng, A. Y., Jordan, M. I., and Russell, S. (2003). Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15*, pages 505–512.

[Yoshii et al., 2006] Yoshii, K., Goto, M., Komatani, K., Ogata, T., , and Okuno, H. G. (2006). Hybrid collaborative and content-based music recommendation using probabilistic model with latent user preferences. In *Proceedings of ISMIR*.

[Zhao and Karypis, 2004] Zhao, Y. and Karypis, G. (2004). Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, 55(3):311–331.

[Zhou et al., 2003] Zhou, D., Bousquet, O., Lal, T., Weston, J., and Sch¨olkopf, B. (2003). Learning with local and global consistency. In *In 18th Annual Conf. on Neural Information Processing Systems*.

[Zhu et al., 2003] Zhu, X., Ghahramani, Z., and Lafferty, J. (2003). Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*.