

Integrating Features from Different Sources for Music Information Retrieval

Tao Li

School of Computer Science
Florida International University
Miami, FL 33199
taoli@cs.fu.edu

Mitsunori Ogihara

Department of Computer Science
University of Rochester
Rochester, NY 14620
ogihara@cs.rochester.edu

Shenghuo Zhu

NEC Laboratories America
Cupertino, CA 95014
zsh@sv.nec-labs.com

Abstract

Efficient and intelligent music information retrieval is a very important topic of the 21st century. With the ultimate goal of building personal music information retrieval systems, this paper studies the problem of identifying “similar” artists using both lyrics and acoustic data. In this paper, we present a clustering algorithm that integrates features from both sources to perform bimodal learning. The algorithm is tested on a data set consisting of 570 songs from 53 albums of 41 artists using artist similarity provided by All Music Guide. Experimental results show that the accuracy of artist similarity classifiers can be significantly improved and that artist similarity can be efficiently identified.

1 Introduction

In *multimedia information retrieval* the data are naturally multi-modal, in the sense that they are represented by multiple sets of features. For example, the representation of a movie has three modes: (i) the personnel (the producer, the director, the editor, the scenario writer, the music composer, the cast, etc.), (ii) the visual features (which summarize the scenarios and the actions), and (iii) the acoustic features (which summarize the voice and the background audio). The representation of popular music is also trimodal in some sense, where the second feature set is replaced by the lyrics. The personnel feature set of the representation of music, however, is significantly smaller than that of movies, since many music artists produce, compose, and perform themselves. This compels one to take the standpoint that the representation of popular music is bimodal, consisting of the acoustic features, which summarize the sound, and the text features, which summarize the words put into the music.

Two fundamental problems in dealing with multime-

dia data are classification and clustering. Classification is the problem of assigning predefined class labels to the data, while clustering is the problem of dividing the data into classes based on their similarity without predefined class labels. These concepts are interchangeably called *supervised learning* and *unsupervised learning*, respectively. Since the proportion of predefined class labels available as part of input is 0% for clustering and 100% for classification, one naturally wonders about the special cases of these two fundamental problems in which only a part of the data has predefined labels. This problem is called *semi-supervised learning*. The main question in semi-supervised learning is whether it is possible to use the unlabeled data to produce something better than the one produced using only the labeled data. In particular, for semi-supervised learning of multi-modal data, i.e., data with heterogeneous sets of features, a natural question is whether multi-modality can be effectively utilized in learning and, if so, whether such multi-modal learning methods produce better results than unimodal methods.

The celebrated paper of Blum and Mitchell [6] is the first to address formally this question. In this paper, Blum and Mitchell study the problem of incorporating unlabeled data in building classifiers in the presence of two feature sets. In particular, they propose a strategy for constructing classifiers called *co-training* for the purpose of making use of unlabeled data. The co-training algorithm proceeds in rounds in the following way: In each round a classifier is built on each of the two feature sets using the current training set, which is initially set to the set of data whose labels are given as input. Then, for each feature set, the point among the unlabeled data for which the classifier with respect to the feature set provides the most confident assertion is selected and is added to the training set of the other feature set along with the assertion. (Note that the two classifiers may select an identical point and disagree on its class label). Blum and Mitchell show that under a certain “independence” assumption about the joint distribution of

the feature sets their co-training algorithm converges in the sense of PAC-learning. Many research efforts have been done for the purpose of extending and generalizing the idea of co-training [1, 9, 16, 25, 28].

It is also possible to design an interactive (or ensemble)¹ learning algorithm (that exploits interactions among classifiers to improve accuracy) for supervised learning (that is, all the data are already labeled). For example, the *co-boosting* algorithm of Collins and Singer [8] uses the individual boosting of the feature sets with the weight adjustments influenced by the labeling of the other classifier(s). The approach can be used not only for supervised learning but for semi-supervised learning (indeed co-boosting algorithm was originally conceived for semi-supervised learning). Although such algorithms may fall into pitfalls due to the highly simple mutual boosting structure, Collins and Singer point out, such multi-modal learning can be very powerful and thus is worth while.

The work of Blum and Mitchell and that of Collins and Singer study the design of effective algorithms multi-modality through interaction for semi-supervised learning and for supervised learning, respectively. This naturally leads to the question of whether multi-modal interactive methods can be more powerful than unimodal methods in the case of unsupervised learning, namely, clustering. The purpose of this paper is to study this question on bimodal clustering (we of course anticipate that bimodal clustering techniques can be naturally extended to general multi-modal clustering). We present a clustering framework for integrating the features based on minimizing disagreement. It is known that in bimodal learning minimizing disagreement between two classifiers can improve the performance of learning [3, 12].

In this paper we present a formalization of the problem of minimizing disagreement in bimodal learning in the Bayesian framework. In the framework, minimizing disagreement can be thought as a simple common theme of multi-modal information retrieval: individual feature sets interact to help each other by reducing disagreement among their outputs. We then present a bimodal clustering algorithm based on the common theme — initialize the cluster layout using the output of the counterpart and try to minimize the disagreement between two modes. We apply the bimodal clustering algorithm to the problem of clustering popular music songs.

The rest of the paper is organized as follows: Section 2 introduces the underlying principle of minimizing the disagreement, Section 3 presents the clustering algorithm of utilizing the general principle, Section 4 describes the two

¹In the literature, the word “interactive” is used often in the case of bimodal learning and “ensemble” in the case of learning with more than two modes. Here we use “interactive” throughout, even to mean learning of data with more than two modes.

heterogeneous feature sets extracted from the lyrics and acoustics data, Section 5 presents the results of experiments. Finally Section 6 concludes.

2 Minimizing the Disagreement

2.1 Theoretical Underpinnings

In this section, we introduce 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.

Our data are bimodal: let X_1 and X_2 be the space of the first mode and the space of the second mode, respectively. Let $X = (X_1, X_2)$ be the product space of X_1 and X_2 . Let 0 and 1 be the class labels of these data, which we will often denote by Y . For each $u \in \{0, 1\}$, we use \bar{u} to its opposite class label, that is, $1 - u$. Suppose that the data in X is subject to a distribution D . Let f be our class label function and let f_1 and f_2 be our class label functions based on the first mode and on the second mode, respectively. The (x) in f and Y are often dropped — we will write $f = u$ to mean $f(x) = u$ and $Y = u$ to mean $Y(x) = u$, etc.

Definition 1 We say that f is a nontrivial classifier if for all $u \in \{0, 1\}$,

$$\Pr(f(x) = u | Y(x) = u) > \Pr(f(x) = \bar{u} | Y(x) = u),$$

where the probability is subject to D . ■

Remark 1 The above nontrivial condition can be restated as $(\forall u \in \{0, 1\})[\Pr(f = u | Y = u) > 1/2]$ and as $(\forall u \in \{0, 1\})[\Pr(f \neq Y) \leq \Pr(f = u)]$.

In [6], it is assumed that x_1 and x_2 are conditionally independent given the labels, i.e.,

$$\Pr(x_1 = x'_1 | x_2 = x'_2) = \Pr(x_1 = x'_1 | f_2(x_2) = f_2(x'_2)).$$

The independence assumption is rather strong, but has been used by many successful applications. Suppose we build hypotheses f'_1 on X_1 and f'_2 on X_2 . Thus, if x_1 and x_2 are conditional independent given the labels, then f'_1 and f'_2 are also conditional independent. The conditional independence of f'_1 and f'_2 can be interpreted as follows:

$$\Pr(f'_1(x_1) = u | f'_2(x_2) = v, Y = y) = \Pr(f'_1(x_1) = u | Y = y)$$

where $u, v, y \in \{0, 1\}$. In other words, *The conditional independence* implies that (i) for all $S_1 \subseteq X_1$ such that the probability of (S_1, X_2) is non-zero, the distribution of X_2 in which the first mode is restricted to S_1 is identical

to the distribution of X_2 with no restriction; and that (ii) for all $S_2 \subseteq X_2$ such that the probability of (X_1, S_2) is non-zero, the distribution of X_1 in which the first mode is restricted to S_2 is identical to the distribution of X_1 with no restriction.

One can show the following (proof omitted):

Theorem 1 *Under conditional independence assumption, the disagreement upper bounds the misclassification error for the nontrivial classifiers.*

In essence, this indicates that, under certain conditions, the disagreement upper bounds the misclassification error. Thus, minimizing disagreement will ideally decrease the upper bound on the misclassification error and could bootstrap the learning algorithm. It should be pointed out that although the principle was originally proved in the context of supervised learning [12], it can be thought as a simple common theme of multi-modal information retrieval: individual feature sets interact to help each other by reducing disagreement among their outputs.

2.2 A Bayesian Framework for Capturing Minimizing Disagreement

Let $x = (x_1, x_2)$ be an observation vector. Then the Bayes decision rule for the first mode is:

$$\Pr(Y = 1|x_1) \leq_1^0 \Pr(Y = 0|x_1).$$

This implies that if the posteriori probability of class 1 (respectively, class 0) given x_1 is larger than the probability of class 0 (respectively, class 1), x_1 is assigned to class 1. Using the Bayes theorem and eliminating the common term $\Pr(x_1)$, we get

$$\Pr(Y = 1) \Pr(x_1|Y = 1) \leq_1^0 \Pr(Y = 0) \Pr(x_1|Y = 0).$$

The Bayes error can be computed as:²

$$\begin{aligned} \epsilon &= \int \min\{\Pr(Y = 1) \Pr(x_1|1), \Pr(Y = 0) \Pr(x_1|0)\} dx_1 \\ &= \Pr(Y = 1) \int_{L_0^1} \Pr(x_1|1) dx_1 + \Pr(Y = 0) \int_{L_1^1} \Pr(x_1|0) dx_1. \end{aligned}$$

Here L_1^1 is the area in which

$$\Pr(Y = 1) \Pr(x_1|Y = 1) > \Pr(Y = 0) \Pr(x_1|Y = 0)$$

and L_0^1 is the area in which

$$\Pr(Y = 1) \Pr(x_1|Y = 1) < \Pr(Y = 0) \Pr(x_1|Y = 0).$$

²We use $\Pr(x_i|j)$ to denote $\Pr(x_i|Y = j)$ where $i = 1, 2$ and $j = 0, 1$.

In other words, if an observation $x_1 \in L_1^1$, it will be classified as in class 1 and if $x_1 \in L_0^1$, it will be classified as in class 0.

Under the conditional independence assumption, the disagreement between two components can be computed as

$$\begin{aligned} E(x_1, x_2) &= \Pr\{(\Pr(Y = 1|x_1) > \Pr(Y = 0|x_1)) \wedge \\ &\quad (\Pr(Y = 1|x_2) < \Pr(Y = 0|x_2))\} \\ &\quad + \Pr\{(\Pr(Y = 1|x_1) < \Pr(Y = 0|x_1)) \wedge \\ &\quad (\Pr(Y = 1|x_2) > \Pr(Y = 0|x_2))\} \\ &= \int_{L_1^1} \int_{L_0^2} p_0(x_1, x_2) + p_1(x_1, x_2) dx_1 dx_2 \\ &\quad + \int_{L_0^1} \int_{L_1^2} p_0(x_1, x_2) + p_1(x_1, x_2) dx_1 dx_2, \end{aligned}$$

where

$$p_0(x_1, x_2) = \Pr(Y = 0) \Pr(x_1|Y = 0) \Pr(x_2|Y = 0),$$

and

$$p_1(x_1, x_2) = \Pr(Y = 1) \Pr(x_1|Y = 1) \Pr(x_2|Y = 1).$$

Here L_1^2 is the region where

$$\Pr(Y = 1) \Pr(x_2|Y = 1) > \Pr(Y = 0) \Pr(x_2|Y = 0)$$

and L_0^2 is the region where

$$\Pr(Y = 1) \Pr(x_2|Y = 1) < \Pr(Y = 0) \Pr(x_2|Y = 0).$$

Similarly, if an observation $x_2 \in L_1^2$, it will be classified as in class 1 and if $x_2 \in L_0^2$, it will be classified as in class 0.

Observe that

$$\begin{aligned} \epsilon &= \Pr(Y = 1) \int_{L_0^1} \Pr(x_1|Y = 1) dx_1 \\ &\quad + \Pr(Y = 0) \int_{L_1^1} \Pr(x_1|Y = 0) dx_1 \\ &= \Pr(Y = 1) \int_{L_0^1} \Pr(x_1|Y = 1) \left(\int \Pr(x_2|Y = 1) dx_2 \right) dx_1 \\ &\quad + \Pr(Y = 0) \int_{L_1^1} \Pr(x_1|Y = 0) \left(\int \Pr(x_2|Y = 1) dx_2 \right) dx_1 \\ &= \int_{L_0^1} \int_{L_0^2} p_1(x_1, x_2) dx_1 dx_2 + \int_{L_0^1} \int_{L_1^2} p_1(x_1, x_2) dx_1 dx_2 \\ &\quad + \int_{L_1^1} \int_{L_0^2} p_0(x_1, x_2) dx_1 dx_2 + \int_{L_1^1} \int_{L_1^2} p_0(x_1, x_2) dx_1 dx_2 \end{aligned}$$

Thus, to ensure that $\epsilon \leq E(x_1, x_2)$, it is sufficient that

$$\int_{L_0^1} \int_{L_0^2} p_1(x_1, x_2) dx_1 dx_2 < \int_{L_1^1} \int_{L_0^2} p_1(x_1, x_2) dx_1 dx_2,$$

and

$$\int_{L_1^1} \int_{L_1^2} p_0(x_1, x_2) dx_1 dx_2 < \int_{L_0^1} \int_{L_0^2} p_0(x_1, x_2) dx_1 dx_2.$$

The above formula can be reduced to

$$Pr(x_1 \in L_0^1 | Y = 1) < Pr(x_1 \in L_1^1 | Y = 1) \quad (1)$$

$$Pr(x_1 \in L_1^1 | Y = 0) < Pr(x_1 \in L_0^1 | Y = 0) \quad (2)$$

The formulas in Eq. (1) and (2) in the above are essentially the same as those in Definition 1 of Section 2. Hence, the disagreement upper bounds can also be derived from the Bayes perspective.

Remark 2 When the conditional independence condition (e.g., equation 1) doesn't hold, to guarantee that disagreement upper bounds the misclassification error, we need

$$Pr(f'_1 = 0 | f'_2 = 0, Y = 1) \leq Pr(f'_1 = 1 | f'_2 = 0, Y = 1)$$

$$Pr(f'_1 = 1 | f'_2 = 1, Y = 0) \leq Pr(f'_1 = 0 | f'_2 = 1, Y = 0)$$

In other words, if

$$Pr(f'_1 \neq Y | f'_2 \neq Y) \leq Pr(f'_1 = Y | f'_2 \neq Y),$$

then the disagreement still upper bounds the misclassification error without the conditional independence condition.

3 Bimodal Clustering

In this section, we present a clustering algorithm that integrates different features based on the principle of minimizing disagreements.

3.1 Measuring Agreements Between Clusterings

Let $D = \{d_1, d_2, \dots, d_n\}$ be a set of n data points. Suppose we are given two clusterings P_1 and P_2 with each consists of a set of clusters:

$$P_i = \{C_i^1, C_i^2, \dots, C_i^{k_i}\}, i = 1, 2$$

where k_i is the number of clusters for clustering P_i , and $D = \bigcup_{j=1}^{k_i} C_i^j$. The first question is how to measure the agreements between the two clusterings.

We use adjusted Rand index to compute the agreement between clusterings. Adjusted Rand Index is a statistic to assess the clustering quality compared against assigned known classes. The Rand Index is defined as the number of pairs of objects which are both located in the same cluster and the same class, or both in different clusters and different classes, divided by the total number of objects [36].

Adjusted Rand Index which adjusts Rand Index is set between $[0, 1]$ [17]. The higher the Adjusted Rand Index, the more resemblance between the two clusterings.

Formally, the adjusted Rand index, *ARI*, is defined as

$$\frac{\sum_{i=1}^{k_1} \sum_{j=1}^{k_2} \binom{n_{ij}}{2} - \sum_{i=1}^{k_1} \binom{n_{i.}}{2} \sum_{j=1}^{k_2} \binom{n_{.j}}{2}}{\frac{\sum_{i=1}^{k_1} \binom{n_{i.}}{2} + \sum_{j=1}^{k_2} \binom{n_{.j}}{2}}{2} - \sum_{i=1}^{k_1} \binom{n_{i.}}{2} \sum_{j=1}^{k_2} \binom{n_{.j}}{2}}.$$

Here n_{ij} denotes the number of objects belonging to both C_i^1 and C_j^2 , $n_{i.} = \sum_{j=1}^{k_2} n_{ij}$, and $n_{.j} = \sum_{i=1}^{k_1} n_{ij}$.

3.2 Clustering Procedure

We present a bimodal clustering approach based on the minimizing the disagreement principle. The algorithm is an extension of the EM method [13]. In each iteration of algorithm, an EM type procedure is employed to bootstrap the model obtained from one data source by starting with the cluster assignments obtained in the previous iteration using the other data source. Upon convergence, the two individual models are used to construct the final cluster assignment. Table 1 listed the notions used for the algorithm and the algorithm procedure is presented in Figure 1.

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
$S = (s_1, \dots, s_n)$	A collection of songs
K	Number of clusters
$\Lambda^1 = (\lambda_1^1, \dots, \lambda_K^1)$	Modal 1 model parameters
$\Lambda^2 = (\lambda_1^2, \dots, \lambda_K^2)$	Modal 2 model parameters
$Y = (y_1, \dots, y_n)$ $y_n \in \{1, \dots, K\}$	Cluster assignment vector
$s \in S$	s represents a song from S
$y_s = k$	Song s is in k -th cluster

Table 1. 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 from the other sources. 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. Other-

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
$$\lambda_k^i = \operatorname{argmax}_{\lambda} \sum_{s: s \in S, y_s = k} \log P(s^i | \Lambda^i)$$
 - 5: **Step III:**
 Sample re-assignment: for each data sample $s \in S$, set
$$y_s = \operatorname{argmax}_k \log P(s^i | \lambda_k^i)$$
 - 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)
-

wise, the algorithm will go back to the allocation step and hopefully get a new clustering.

4 Two Heterogeneous Feature Sets

We address the issue of identifying the artist style using both content and lyrics. Ellis et al. [26] 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. Recently, [35] shows the usefulness of multi-modal learning for music artist style classification. In this section, we describe the feature sets extracted from the lyrics and the acoustic content.

4.1 Text-Based Style Features

Recently, there has appeared some work that exploits the use of non-sound information for music information retrieval. Whitman and Smaragdīs [35] study the use of the descriptions (obtained from All Music Guide) and the sounds of artists together to improve classification. Whitman, Roy, and Vercoe [34] show that the meanings the artists associate with words can be learned from

the sound signals. A number of researchers also presented probabilistic approaches to model music and text jointly [5, 7, 22, 29]. From these results, it can be hypothesized that by analyzing how words are used to generate lyrics, artists can be distinguished from others and similar artists can be identified.

Previous study on stylometric analysis has shown that statistical analysis on text properties could be used for text genre identification and authorship attribution [2, 18, 30] and over one thousand stylometric features (style makers) have been proposed in variety research disciplines [32]. To choose features for analyzing lyrics, one should be aware of the characteristics of popular song lyrics. For instance, song lyrics are usually brief and are often built from a very small vocabulary. In song lyrics, words are uttered with melody, so the sound they make plays an important role in determination of words. The stemming technique, though useful in reducing the number of words to be examined, may have a negative effect. In song lyrics, word orders are often different from those in conversational sentences and song lyrics are often presented without punctuation.

To account for 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. We did not apply stemming operations.
- *Part-of-Speech statistics:* We also use the output of Brill's part-of-speech (POS) tagger [4] as the basis for feature extraction. POS statistics usually reflect the characteristics of writing. There are 36 POS features extracted for 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 features of individual word-tokens in the text. The most basic lexical features are lists of 303 generic function words taken from [23]³, 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

³Available on line at <http://www.cse.unsw.edu.au/~min/ILLDATA/Function.word.htm>

placement, and word length distribution as our features. Word orders and lengths are very useful since the writing of lyrics usually follows certain melody.

4.2 Content-Based Features

There has been a considerable amount of work in extracting descriptive features from music signals for music genre classification and artist identification [14, 19, 27, 33, 21]. In our study, we use timbral features along with wavelet coefficient histograms. The feature set consists of the following three parts and totals 35 features.

4.2.1 Mel-Frequency Cepstral Coefficients (MFCC)

MFCC is a feature set popular in speech processing and is designed to capture short-term spectral-based features. To obtain the feature, we first compute, for each frame, the logarithm of the amplitude spectrum based on short-term Fourier transform, where the frequencies are divided into thirteen bins using the Mel-frequency scaling. (The “cepstrum” is the name coined for this logarithm.) After taking the logarithm of the amplitude spectrum, the frequency bins are grouped and smoothed according to Mel-frequency scaling, which is design to agree with perception. MFCC features are generated by decorrelating the Mel-spectral vectors using discrete cosine transform. In this study, we use the first five bins, and compute the mean and variance of each over the frames.

4.2.2 Short-Term Fourier Transform Features (FFT)

This is a set of features related to timbral textures and is not captured using MFCC. It consists of the following five types. More detailed descriptions can be found in [33].

Spectral Centroid is the centroid of the magnitude spectrum of short-term Fourier transform and is a measure of spectral brightness. *Spectral Rolloff* is the frequency below which 85% of the magnitude distribution is concentrated. It measures the spectral shape. *Spectral Flux* is the squared difference between the normalized magnitudes of successive spectral distributions. It measures the amount of local spectral change. *Zero Crossings* is the number of time domain zero crossings of the signal. It measures noisiness of the signal. *Low Energy* is the percentage of frames that have energy less than the average energy over the whole signal. It measures amplitude distribution of the signal.

We compute the mean for all five types and the variance for all but zero crossings.

4.2.3 Daubechies Wavelet Coefficient Histograms (DWCH)

Daubechies wavelet filters are ones that are popular in image retrieval (see [10]). To extract DWCH features, the db_8 filter with seven levels of decomposition is applied to thirty seconds of sound signals. After the decomposition, 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 used [11, 21] to approximate the probability distribution at each subband. In addition, the subband energy, defined as the mean of the absolute value of the coefficients, is also computed at each subband. A few trials reveal that of the seven subbands of db_8 (1: 11025–22050 Hz, 2: 5513–11025Hz, 3: 2756–5513Hz, 4: 1378–2756Hz, 5: 689–1378Hz, 6: 334–689Hz, 7: 0–334Hz), subbands 1, 2, and 4 show little variation. We thus choose to use only the remaining four subbands, 3, 5, 6, and 7, for our experiments. In fact, The subbands match the models of sound octave-division for perceptual scales [20].

5 Experiments

In this section, we perform experiments to evaluate whether the clustering algorithms based on minimizing disagreement can be more powerful than unimodal methods.

5.1 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 clusters are listed in Table 2. 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.

5.2 Evaluation Measures

As discussed above, we use the cluster structures obtained from All Music Guide as labels to evaluate the clustering performance. We use Purity, Entropy and Accuracy [38] as our performance measures. We expect these

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

Table 2. Cluster Memberships.

measures would provide us with good insights on how our algorithm works.

Purity measures the extent to which each cluster contained data points from primarily one class [38]. The purity of a clustering solution is obtained as a weighted sum of individual cluster purity values and is given by

$$Purity = \sum_{i=1}^K \frac{n_i}{n} P(S_i), P(S_i) = \frac{1}{n_i} \max_j (n_i^j),$$

where S_i is a particular cluster of size n_i , n_i^j is the number of documents of the i -th input class that were assigned to the j -th cluster, K is the number of clusters and n is the total number of points⁴. In general, the larger the values of purity, the better the clustering solution is.

Entropy measures how classes distributed on various clusters [38]. The entropy of the entire clustering solution is computed as:

$$Entropy = -\frac{1}{n \log_2 m} \sum_{i=1}^K \sum_{j=1}^m n_i^j \log_2 \frac{n_i^j}{n_i},$$

where m is the number of original labels, K is the number of clusters. Generally, the smaller the entropy value, the better the clustering quality is.

Accuracy discovers the one-to-one relationship between clusters and classes, therefore to measure the extent to which each cluster contained data points from the corresponding class. It sums up the whole matching degree between all pair class-clusters. Accuracy of the clustering can be represented as:

$$Accuracy = \frac{\max(\sum_{C_k, L_m} T(C_k, L_m))}{N},$$

where C_k denotes the k -th cluster, and L_m is the m -th class. $T(C_k, L_m)$ is the number of entities which belong to class m are assigned to cluster k . Accuracy computes

⁴ $P(S_i)$ is also called the individual cluster purity.

the maximum sum of $T(C_k, L_m)$ for all pairs of clusters and classes, and these pairs have no overlaps. The larger accuracy usually means the better clustering performance.

5.3 Experimental Comparisons

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:

- **Feature-Level Integration:** Feature-level integration performs K-means clustering after simply concatenating the features obtained from the two data sources.
- **Cluster Integration:** Cluster integration refers to the procedure of obtaining a combined clustering from multiple clusterings of a dataset [31, 24, 15]. Formally, let $C_1^1, \dots, C_1^{k_1}$ denote the clusters obtained from source 1, and $C_2^1, \dots, C_2^{k_2}$ denote the clusters obtained from source 2. Each point d_i can be represented as a $(k_1 + k_2)$ -dimensional vector

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

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

A combined clustering can be found by applying the K-means algorithm on the new representation.

- **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, \dots, 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:

1. Sequential Integration I: first cluster based on content, then integrate with lyrics;
2. Sequential Integration II: first cluster based on lyrics, then integrate with content.

Figure 1 illustrates and summarizes various strategies for integrating different information sources.

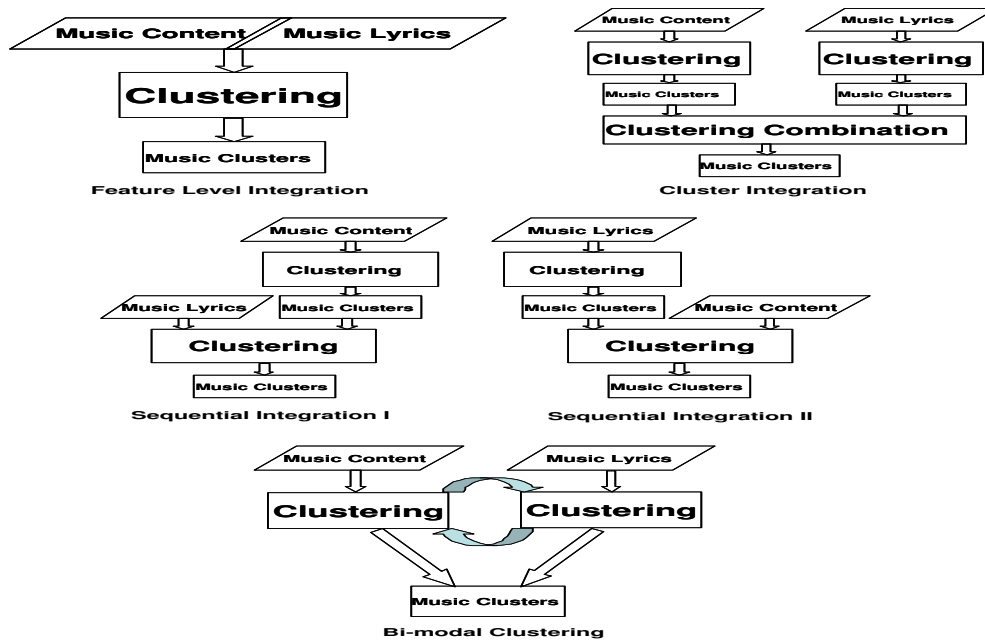


Figure 1. Various strategies for integrating different information sources.

5.4 Analysis of the Results

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 3 presents the experimental results.

Feature Set(s)	Purity	Entropy	Accuracy
Content-only	0.436	0.731	0.438
Lyrics-only	0.444	0.728	0.402
Feature-Level Integration	0.425	0.729	0.380
Cluster Integration	0.465	0.725	0.423
Sequential Integration I	0.431	0.724	0.434
Sequential Integration II	0.438	0.734	0.407
bimodal Clustering	0.471	0.697	0.453

Table 3. Performance Comparison. The numbers are obtained by averaging over ten trials.

From the table, we observe the following:

- The performance of purity, entropy, and accuracy relative to the other is not always consistent in our comparison, i.e., higher purity values do not necessarily correspond to lower entropy values, or to higher accuracy values. This is because different evaluation measures consider different aspects of the clustering results. For example, the entropy measure takes into

account the entire distribution of the data in a particular cluster and not just the largest class as in the computation of the purity. The accuracy considers the relationships among all pair class-clusters. We compare these three different measures and hope they would provide enough insights for our experiments.

- The purity and accuracy of feature-level integration are worse than those of content-only and lyric-only clustering methods, while their entropy values are fairly close. 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 [37].
- Cluster Integration: The cluster integration performs better than content-only and lyrics-only: cluster integration have higher purity and accuracy values and lower entropy values than those of content-only and lyrics-only. This actually conforms to the results in [15]: cluster aggregation would usually provide better clustering results.
- 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.
- Our bimodal clustering outperforms all other methods in all three performance measures. The bimodal

clustering algorithm can be thought of 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.

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 don't. 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.

To investigate the relationship between the clustering performance and the agreement with respect to the two sources, we take a closer look at our experiments. Figure 2 shows the cluster performance (entropy and purity values) and the (dis-)agreements between two sources in a trial. Each unit on the X-axis represents five iterations of the algorithm and the Y-axis shows the performance value. We can observe from Figure 2 that as the agreement between the two sources increases, the clustering quality also tends to increase (i.e., entropy is generally decreasing while purity is increasing).

6 Conclusion

In this paper, we study the problem on whether multimodal interactive methods can be more powerful than unimodal methods in the case of clustering. In particular, we present a clustering framework for integrating the features based on minimizing disagreement. Experimental results on a data set consisting of 570 songs from 41 artists of 53 albums show the effectiveness of our approach.

There are two natural avenues for future research. The first natural direction is on music annotation. How can we automatically and efficiently generate music style or similarity information? Note we did not agree completely with the artist similarity obtained from All Music Guide, but nonetheless used it as the ground truth to evaluate our algorithms in the experiments. Can we incorporate the opinions

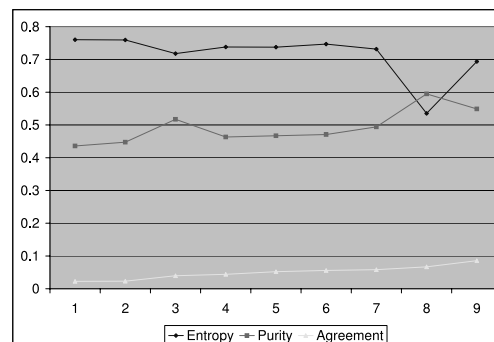


Figure 2. Relationships Between Clustering Performance and Agreements. Each unit on the X-axis represents 5 iterations of the algorithm and the Y-axis shows the performance value.

from music experts or take into account the views from individual users? Second, it would also be interesting to extend the bimodal algorithm by using statistical inference techniques to adaptively weight different data sources during the clustering process.

7 Acknowledgements

This work is supported in part by NSF Career Award IIS-054680, NSF grants EIA-0080124, and EIA-0205061.

References

- [1] Steven Abeny. Bootstrapping. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 360–367. Morgan Kaufmann Publishers, 2002.
- [2] Shlomo Argamon, Marin Saric, and Sterling S. Stein. Style mining of electronic messages for multiple authorship discrimination: first results. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 475–480. ACM Press, 2003.
- [3] Suzanna Becker. Mutual information maximization: Models of cortical self-organization. *Network: Computation in Neural Systems*, 7(1):7–31, February 1996.
- [4] Eric Bill. Some advances in transformation-based parts of speech tagging. In *Proceedings of the twelfth national conference on Artificial intelligence (vol. 1)*, pages 722–727, 1994.
- [5] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.
- [6] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory (COLT'98)*, pages 92–100. ACM Press, 1998.

- [7] Eric Brochu and Nando de Freitas. Name that song!: A probabilistic approach to querying on music and text. In *Neural Information Processing Systems: Natural and Synthetic*, 2002.
- [8] Michael Collins and Yoram Singer. Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.
- [9] Sanjoy Dasgupta, Michael L. Littman, and David McAllester. PAC generalization bounds for co-training. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 375–382, Cambridge, MA, 2002.
- [10] I. Daubechies. *Ten lectures on wavelets*. SIAM, Philadelphia, 1992.
- [11] Afshin David and Sethurman Panchanathan. Wavelet-histogram method for face recognition. *Journal of Electronic Imaging*, 9(2):217–225, 2000.
- [12] Virginia R. De Sa and Dana Ballard. Category learning through multi-modality sensing. *Neural Computation*, 10(5):1097–1117, 1998.
- [13] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1, 38 1977.
- [14] Jonathan Foote and Shingo Uchihashi. The beat spectrum: a new approach to rhythm analysis. In *IEEE International Conference on Multimedia & Expo 2001*, 2001.
- [15] Aristides Gionis, Heikki Mannila, and Panayiotis Tsaparas. Clustering aggregation. In *ICDE*, pages 341–352, 2005.
- [16] Sally Goldman and Yan Zhou. Enhancing supervised learning with unlabeled data. In *Proceedings of the 17th International Conference on Machine Learning (ICML'00)*, pages 327–334, 2000.
- [17] Milligan GW and Cooper MC. A study of the comparability of external criteria for hierarchical cluster analysis. *Multivar Behav Res.* 21:846–850, 1986.
- [18] Brett Kessler, Geoffrey Nunberg, and Hinrich Schütze. Automatic detection of text genre. In Philip R. Cohen and Wolfgang Wahlster, editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 32–38, 1997.
- [19] Jean Laroche. Estimating tempo, swing and beat locations in audio recordings. In *Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA01)*, 2001.
- [20] Guohui Li and Ashfaq A. Khokhar. Content-based indexing and retrieval of audio data using wavelets. In *IEEE International Conference on Multimedia and Expo (II)*, pages 885–888, 2000.
- [21] Tao Li, Mitsunori Ogihara, and Qi Li. A comparative study on content-based music genre classification. In *Proceedings of 26th Annual ACM Conference on Research and Development in Information Retrieval (SIGIR 2003)*, pages 282–289. ACM Press, 2003.
- [22] Beth Logan, Patrawadee Prasangit, and Pedro Moreno. Fusion of semantic and acoustic approaches for spoken document retrieval. In *Proceedings of ISCA Workshop on Multilingual Spoken Document Retrieval*, 2003.
- [23] Roger Mitton. Spelling checkers, spelling correctors and the misspellings of poor spellers. *Information Processing and Management*, 23(5):103–209, 1987.
- [24] Stefano Monti, Pablo Tamayo, Jill Mesirov, and Todd Gloub. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning Journal*, 52(1-2):91–118, 2003.
- [25] Kamal Nigam and Rayid Ghani. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the 2000 ACM CIKM International Conference on Information and Knowledge Management (CIKM'00)*, pages 86–93. ACM Press, 2000.
- [26] Daniel P.W.Ellis, Brian Whitman, Adam Berenzweig, and Steve Lawrence. The quest for ground truth in musical artist similarity. In *Proceedings of 3rd International Conference on Music Information Retrieval*, pages 170–177, 2002.
- [27] L. Rabiner and B.H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, NJ, 1993.
- [28] Dan Roth and Dmitry Zelenko. Toward a theory of learning coherent concepts. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI'00)*, pages 639–644, 2000.
- [29] Malcolm Slaney. Semantic-audio retrieval. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2002.
- [30] Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. Automatic text categorization in terms of genre and author. *Computational Linguistics*, 26(4):471–496, 2000.
- [31] Alexander Strehl and Joydeep Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, March 2003.
- [32] Fiona J. Tweedie and R. Harald Baayen. How variable may a constant be? Measure of lexical richness in perspective. *Computers and the Humanities*, 32:323–352, 1998.
- [33] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5), July 2002.
- [34] B. Whitman, D. Roy, and B. Vercoe. Learning word meanings and descriptive parameter spaces from music. In *Proceedings of the HLT-NAACL03 workshop on learning wording meaning from non-linguistic data*, 2003.
- [35] B. Whitman and P. Smaragdis. Combining musical and cultural features for intelligent style detection. In *Proceedings of 3rd International Conference on Music Information Retrieval*, pages 47–52, 2002.
- [36] Rand WM. Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc*, 66:846–850, 1971.
- [37] Lizhong Wu, Sharon L. Oviatt, and Philip R. Cohen. Multimodal integration - a statistical view. *IEEE Transactions on Multimedia*, 1(4):334–341, 1999.
- [38] Ying Zhao and George Karypis. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, 55(3):311–331, 2004.