LEARNING THE SIMILARITY OF AUDIO MUSIC IN BAG-OF-FRAMES REPRESENTATION FROM TAGGED MUSIC DATA

Ju-Chiang Wang[1,2], Hung-Shin Lee[1,2], Hsin-Min Wang[2] and Shyh-Kang Jeng[1]
[1] Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan
[2] Institute of Information Science, Academia Sinica, Taipei, Taiwan
{asriver, hslee, whm}@iis.sinica.edu.tw, skjeng@cc.ee.ntu.edu.tw

ABSTRACT

Due to the cold-start problem, measuring the similarity between two pieces of audio music based on their low-level acoustic features is critical to many Music Information Retrieval (MIR) systems. In this paper, we apply the bag-of-frames (BOF) approach to represent low-level acoustic features of a song and exploit music tags to help improve the performance of the audio-based music similarity computation. We first introduce a Gaussian mixture model (GMM) as the encoding reference for BOF modeling, then we propose a novel learning algorithm to minimize the similarity gap between low-level acoustic features and music tags with respect to the prior weights of the pre-trained GMM. The results of audio-based query-by-example MIR experiments on the MajorMiner and Magnatagatune datasets demonstrate the effectiveness of the proposed method, which gives a potential to guide MIR systems that employ BOF modeling.

1. INTRODUCTION

Measuring the similarity between two pieces of music is a fundamental but difficult task in Music Information Retrieval (MIR) research [1] since music similarity is inherently based on human subjective point of view and can be bias among people who have different musical tastes and prior knowledge. A piece of music contains a variety of musical contents, including the low-level audio signal; the metadata, such as the artist, album, song name, and release year; and a number of high-level perceptive descriptions, such as timbre, instrumentation, style/genre, mood, and social information (e.g., tags, blogs, and explicit or implicit user feedback). Among the musical contents, only the audio signal is always available while the metadata and high-level perceptive descriptions are often unavailable or expensive to obtain. Owing to the cold-start problem, measuring the similarity between two pieces of audio music based on their low-level acoustic features is critical to many MIR systems [2, 3]. These systems are usually evaluated against the objective criteria derived from the metadata and high-level perceptive descriptions, which in fact correspond to the subjective criteria that humans use to measure music similarity. The similarity gap between the acoustic features and human subjective perceptions inevitably degrades the performances of the MIR systems. The gap may come from an insufficient song-level acoustic feature extraction or representation and an ill similarity metric. Therefore, the goal of improving audio-based music similarity computation is to reduce the gap between audio features and human perceptions, and it can be achieved from a music feature representation perspective [3-8] or a similarity learning perspective [1, 10].

Due to the "glass ceiling" of performance that the pure audio-based music similarity computation systems have faced, several high-level perceptive descriptions, which are considered having a smaller gap between the similarity computed on them and the subjective similarity of human, have been employed in some previous work. For example, in [6, 7], an intermediate semantic space (e.g. genre or text caption) is used to bridge and reduce the similarity gap. During recent years, social information has been very popular and become a major source of contextual knowledge for MIR systems. The social information generated by Internet users makes the "wisdom of crowds" available for investigating the general criteria of human subjective music similarity. In [1], the music blogs are exploited to learn the music similarity metric of audio features. In [8], the social tags are concatenated with the audio features to represent music in a query-by-example MIR scenario. Furthermore, Kim et al. [9] conduct explicit and implicit user feedback, which can be implemented by collaborative filtering (CF, the user-artist matrix), to measure artist similarity. Surprisingly, the experimental results show that CF can be a very efficient source in music similarity computation. Afterwards, the CF data is used in [10] to learn the audio-based similarity metric and significant improvements in query-by-example MIR performance are achieved with three types of song-level representations, namely, acoustic, auto-tag, and human-tag representations.

This work was supported in part by the Taiwan e-Learning and Digital Archives Program (TELDAP) sponsored by the National Science Council of Taiwan under Grant: NSC 100-2631-H-001-013.

In the abovementioned work, music tags are mostly treated as part of music features to represent a song [8-10]. In this paper, we adopt music tags to create a ground truth semantic space to be used to measure human subjective similarity for three reasons. First, music tags are human labels that represent human musical perceptions. According to previous studies [9, 10], the similarity from tags is highly relative to the subjective similarity for evaluation, i.e., the similarity gap is relatively small. Second, music tags are free-text labels that include all kinds of musical information, such as genre, mood, instrumentation, personal preference, and metadata, which are used to objectively evaluate the effectiveness of audio-based similarity computation in previous work. Third, music tags are generally considered noisy, redundant, bias, and unstable when collected from a completely non-constrained tagging environment, such as last.fm. Consequently, several web-based music tagging games have been created with a purpose of collecting reliable and useful tags, e.g., MajorMiner.org [12] and Tag A Tune [13]. In these tagging games, music clips are randomly assigned to taggers in order to reduce the tagging bias. Carefully extracting tags with high term frequencies and merging equivalent tags can intuitively reduce the noisy factors. With a set of well-refined music tags, the semantic space which simulates the human music similarity can be established.

In most audio-based MIR systems, the sequence of short-time frame-based or segment-based acoustic feature vectors of a song is converted into a fixed-dimensional vector so that the song-level semantic descriptions (or tags) can be incorporated into it. The bag-of-frames (BOF) or bag-of-segments approach is a popular and efficient way to represent a set of frame-based acoustic vectors of a song and has been widely used in MIR applications [8,10,14]. In the traditional BOF approach, a set of frame representatives (e.g., codebook, denoted as an encoding reference hereafter) are selected or learned in an unsupervised manner, then a song is represented by the histogram over the encoding reference.

In the BOF representation vector, each dimension represents the effective quantity of its corresponding frame representative (e.g., codeword) within a song. Based on the effective quantities, we can estimate the audio-based similarity of two songs. Motivated by the metric learning for audio-based music similarity computation in [1, 10], we could learn a metric transformation over the BOF representation vector by minimizing the similarity gap between acoustic features and music tags. Since the BOF vector is generated by the encoding reference, the minimization of similarity gap can be achieved by learning the encoding reference rather than learning a metric transformation on the native BOF space. This leads to a supervised method for learning the encoding reference from a tagged music dataset to improve the BOF representation. Hopefully, the learned encoding reference could better generalize the BOF modeling than a stacking transformation over the native metric.

The remainder of this paper is organized as follows. Section 2 describes the audio feature extraction module and song-level BOF representation. In Section 3, we introduce the method for learning the encoding reference from the tagged music data. In Section 4, we evaluate the proposed method on the MajorMiner and Magnatagatune datasets in a query-by-example MIR scenario. Finally, we summarize our conclusions in Section 5.

2. BAG-OF-FRAMES REPRESENTATION FOR ACOUSTIC FEATURES

2.1 Frame-based Acoustic Feature Extraction

We use MIRToolbox 1.3 for acoustic feature extraction [14]. As shown in Table 1, we consider four types of features, namely, dynamic, spectral, timbre, and tonal features. To ensure alignment and prevent mismatch of different features in a vector, all the features are extracted with the same fixed-sized short-time frame. Given a song, a sequence of 70-dimensional feature vectors is extracted with a 50ms frame size and 0.5 hop shift. Then, we normalize the 70-dimensional frame-based feature vectors in each dimension to mean 0 and standard deviation 1.

Types	Feature Description	Dim
dynamic	rms	1
spectral	centroid, spread, skewness, kurtosis, entropy, flatness, rolloff 85, rolloff 95, brightness, roughness, irregularity	11
timbre	zero crossing rate, spectral flux, MFCC, delta MFCC, delta-delta MFCC	41
tonal	key clarity, key mode possibility, HCDF, chroma , chroma peak, chroma centroid	17

Table 1. The music features used in the 70-dimensional frame-based music feature vector.

2.2 The Encoding Reference and BOF Representation

The BOF approach is argued that each frame of a song should not be treated equally, and an isolated frame of low-level acoustic feature is not representative for high-level perceptive descriptions [15]. Besides, the effectiveness of BOF modeling is highly impacted by the size of encoding reference and will encounter a glass ceiling when the size is too large [16]. Our goal of improving the encoding reference for BOF modeling is twofold: First, we aim at choosing a type of frame representative that gives better generalization ability and a more reliable distance measure criterion. Second, each frame representative should not have equal information load during song-level encoding.

The BOF modeling starts with generating the encoding reference from a set of available frames (denoted as $\boldsymbol{F}$). The frames are usually selected randomly and uniformly from each song in a music dataset. We use a Gaussian mixture model (GMM) instead of a codebook derived by the *K*-mean algorithm as the encoding reference [17]. In the

GMM, each component Gaussian distribution, denoted as z_k, k=1,…,K, corresponds to a frame representative. The GMM is trained on $\boldsymbol{F}$ by the expectation-maximization (EM) algorithm, and is expressed as follows:

$$p(\mathbf{v}) = \sum_{k=1}^{K} \pi_k N_k(\mathbf{v} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad (1)$$

where $\mathbf{v}$ is a frame-based feature vector, $N_k(\cdot)$ is the k-th component Gaussian distribution with mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$, and π_k is the prior weight of the k-th mixture component. Given $\mathbf{v}$, the posterior probability of a mixture component is computed by:

$$p(z_k \mid \mathbf{v}) = \frac{\pi_k N_k(\mathbf{v} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{m=1}^{K} \pi_m N_m(\mathbf{v} \mid \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}. \quad (2)$$

Given a song s with L frames, its BOF posterior-probability representation (denoted as vector $\mathbf{x}$) is computed by:

$$x_k \leftarrow p(z_k \mid s) = \frac{1}{L} \sum_{t=1}^{L} p(z_k \mid \mathbf{v}_t), \quad (3)$$

where x_k is the k-th element in vector $\mathbf{x}$. When encoding a frame by GMM, the posterior probability is based on the likelihood of each component Gaussian distribution. The posterior probability of each mixture component yields a soft-assigned encoding criterion which enhances the modeling ability of the GMM-based encoding reference over the vector-quantization-based (VQ-based) one.

Our contention is that the diversity of frame representatives in the encoding reference is proportional to the ability of the BOF modeling, i.e., the BOF modeling can involve more audio information of a song to be encoded. However, like other encoding references (e.g., a set of randomly selected vectors or a trained codebook), the GMM is generated in an unsupervised manner. The factors that we can control includes the number of components in GMM, i.e., K, the types of acoustic features used in the frame-based vector, and the construction of $\boldsymbol{F}$. Except for K, the other two factors are fixed in the beginning. As K increases, the frame representatives become more diverse, but some of them are in fact redundant. This motivates us to determine the importance of each frame representative in a discriminative way. The EM training for GMM provides the estimation of the data distribution over $\boldsymbol{F}$, which is assumed to follow a mixture of Gaussian distributions, by the maximum likelihood criterion. The prior π_k of the k-th component Gaussian represents the corresponding effective number of frames in training set $\boldsymbol{F}$. However, the construction of $\boldsymbol{F}$ implies that the estimated distribution of $\boldsymbol{F}$ actually does not have information about the song-level distribution of acoustic feature vectors. In other words, it may not reflect the importance of each mixture component when encoding a song. In fact, as will be discussed later in Sec. 4, our experimental results show that setting the trained priors to a uniform distribution improves the MIR performance.

In light of the observations described above and the beneficial characteristics of music tags, we readily incorporate the tagged music data as a supervision guide to determine the importance of each mixture component in the GMM.

3. LEARNING THE AUDIO-BASED SIMILARITY

In this work, learning the similarity of audio music from tagged music data is achieved by learning the encoding reference to minimize the similarity gap between low-level acoustic features and high-level music tags. To this end, we conduct learning with respect to the parameters of the GMM trained from $\boldsymbol{F}$. In this paper, we only consider the relearning of the prior probabilities, i.e., the pre-learned parameters $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$, k=1,…,K, are fixed. The proposed iterative learning algorithm has two steps, namely, encoding songs into BOF vectors and minimizing the similarity gap with respect to the prior probabilities of the GMM.

3.1 Preliminary

Suppose there is a tagged music corpus $\boldsymbol{D}$ with N songs. Given a song s_i in $\boldsymbol{D}$, we have its BOF vector $\mathbf{x}_i \in \boldsymbol{R}^{K \times 1}$, which is encoded by the GMM to represent the acoustic features, and its tag vector $\mathbf{y}_i \in \{0,1\}^{M \times 1}$, in which each tag is *binary* labeled (multi-label case) from a pre-defined tag set with M tags. Two similarity matrices are defined: S_X is computed on the N BOF vectors, and S_Y is computed on the N tag vectors. We estimate the acoustic similarity between s_i and s_j in $\boldsymbol{D}$ by computing the inner product of $\mathbf{x}_i$ and $\mathbf{x}_j$. Therefore, the acoustic similarity matrix S_X of $\boldsymbol{D}$ can be expressed as:

$$S_X = \mathbf{X}^T \mathbf{X}, \quad (4)$$

where $\mathbf{X}$ is a K-by-N matrix with $\mathbf{x}_i$ as its i-th column. The tag similarity matrix S_Y of $\boldsymbol{D}$ is expressed as:

$$S_Y = \mathbf{Y}^T \mathbf{Y}, \quad (5)$$

where $\mathbf{Y}$ is an M-by-N matrix with $\mathbf{y}_i / \|\mathbf{y}_i\|$ as its i-th column. Since each song may have different numbers of tags, to ensure that the tag-based similarity of a song itself is always the largest, we compute the cosine similarity between $\mathbf{y}_i$ and $\mathbf{y}_j$ in Eq. (5) to estimate the tag-based similarity to simulate the human similarity between s_i and s_j.

The methods for audio-based similarity computation can be evaluated by a query-by-example MIR system, i.e., given a query song with the audio signal only, the system ranks all the songs in the database based on audio-based similarity computation only. To evaluate the effectiveness of S_X, we perform *leave-one-out* MIR tests to evaluate the normalized discounted cumulative gain (NDCG) [18] with respect to the ground truth relevance derived by S_Y. That is, each song s_i in $\boldsymbol{D}$ is taken as a query song in turn, the output ranked list for s_i is generated by sorting the elements in the i-th row of S_X in descending order, and the corresponding ground truth relevance is the i-th row of S_Y. The

$\text{NDCG}_i@P$, which represents the quality of ranking of the top P retrieved songs for query s_i, is formulated as follows:

$$\text{NDCG}_i @ P = \frac{1}{Q_P}\left\{ R_i(1) + \sum_{j=2}^{P} \frac{R_i(j)}{\log_2 j} \right\}, \quad (6)$$

where $R_i(j)$ is the ground truth relevance (obtained from the i-th row of S_Y) of the j-th song on the ranked list, and Q_P is the normalization term representing the ideal ranking of the P songs [18]. Intuitively, if more songs with large ground truth relevance are ranked higher, a larger NDCG will be obtained. The query-by-example MIR performance on $\boldsymbol{D}$ based on S_X with respect to S_Y is evaluated by

$$\text{NDCG}(\boldsymbol{D})@P = \frac{1}{N}\sum_{i=1}^{N} NDCG_i @ P. \quad (7)$$

The larger NDCG in Eq. (7) is, the more effective the audio similarity computation for S_X is.

3.2 Minimizing the Similarity Gap

We define a K-by-K symmetric transformation matrix $\mathbf{W}$ for the BOF vector space. The transformed BOF vector for s_i is expressed by $\mathbf{W}\mathbf{x}_i$, and the new acoustic similarity matrix S_T of $\boldsymbol{D}$ can be obtained by:

$$S_T = (\mathbf{W}\mathbf{X})^T(\mathbf{W}\mathbf{X}) = \mathbf{X}^T\mathbf{T}\mathbf{X}, \quad (8)$$

where $\mathbf{T}=\mathbf{W}^T\mathbf{W}$. Therefore, minimizing the similarity gap between the transformed BOF vector space and human tag vector space is equivalent to minimizing the distance or maximizing the correlation [19] between the two kernel matrices S_T and S_Y with respect to $\mathbf{W}$. In this paper, motivated by the work in [20], we express the N songs in $\boldsymbol{D}$ as two random vectors, $\mathbf{Z}_x \in \boldsymbol{R}^{N\times 1}$ for the transformed acoustic feature and $\mathbf{Z}_y \in \boldsymbol{R}^{N\times 1}$ for the tag label, which follow two multivariate Gaussian distributions N_x and N_y, respectively. There exists a simple bijection between the two multivariate Gaussians. Without loss of generality, we assume N_x and N_y have an equal mean and are parameterized by ($\boldsymbol{\mu}$, S_T) and ($\boldsymbol{\mu}$, S_Y), respectively. Then, the "closeness" between N_x and N_y can be measured by the relative entropy $\text{KL}(N_x \| N_y)$ (i.e., the KL-divergence), which is equivalent to $d(S_T \| S_Y)$:

$$d(S_T \| S_Y) = \frac{1}{2}\left\{ tr(S_T S_Y^{-1}) - \log | S_T S_Y^{-1} | - N \right\}, \quad (9)$$

where $tr(\cdot)$ and $|\cdot|$ are the trace and determinant of a matrix, respectively. The minimization of $d(S_T \| S_Y)$ can be solved by setting the derivative of $d(S_T \| S_Y)$ with respect to $\mathbf{T}$ to zero. The solution that minimizes $d(S_T \| S_Y)$ is as follows:

$$\mathbf{T}^* = \left(\mathbf{X}(S_Y)^{-1}\mathbf{X}^T\right)^{-1}. \quad (10)$$

Since $\mathbf{W}$ is symmetric, the optimal matrix $\mathbf{W}$ is derived by

$$\mathbf{W}^* = (\mathbf{T}^*)^{1/2}. \quad (11)$$

To prevent singularity, a small value 0.001 is added to each diagonal element of the matrices that are inversed in solving $\mathbf{W}$. If we restrict $\mathbf{W}$ in Eq. (8) to be diagonal, i.e., we ignore the correlation among different dimensions in the BOF vector, and define vector $\mathbf{w} \equiv \text{diag}(\mathbf{W})$, the optimal $\mathbf{w}^*$ is the diagonal of $\mathbf{W}^*$:

$$\mathbf{w}^* = \text{diag}(\mathbf{W}^*), \quad (12)$$

where each element in $\mathbf{w}^*$ must be greater than zero. The derivations of Eqs. (10) and (12) are skipped due to the space limitation.

In the testing phase, each song is first encoded into a BOF vector by the GMM using Eq. (3). Then, the audio-based similarity between any two songs s_i and s_j is computed as $\mathbf{x}_i^T\mathbf{T}^*\mathbf{x}_j$, where $\mathbf{T}^*$ can be a full or diagonal matrix according to the initial setting of $\mathbf{W}$ in Eq. (8). In the experiments, this method with full transformation and diagonal transformation is denoted as FullTrans and DiagTrans respectively, while the method without transformation is denoted as OrigGMM (i.e., the native GMM).

3.3 Relearning the Priors of the GMM

Instead of learning a transformation matrix, we can also minimize the similarity gap by relearning the prior weights of the GMM. We propose a two-step iterative learning method, which iteratively updates the prior weights of the GMM until convergence. The NDCG in Eq. (7) can be used as the criterion for checking the convergence of the learning procedure. The minimization of similarity gap implies the improvement in NDCG since the learned S_T tries to preserve the structure of S_Y, which is used as the ground truth relevance in computing NDCG. If NDCG is no longer improved, the learning algorithm stops. The learning method is summarized in Algorithm 1.

According to Algorithm 1, there are two steps in an iteration. Line 05 corresponds to the first step, which encodes all songs into their BOF vectors; and lines 11 and 13 correspond to the second step, which minimizes the similarity gap with respect to the prior weights of the GMM. Since encoding all songs in $\boldsymbol{D}$ is a complicated procedure, directly optimizing NDCG with respect to the parameters of the GMM with Eqs. (1), (2) and (3) is infeasible. Therefore, we turn to find an indirect solution that minimizes the similarity gap with respect to the priors of the GMM. We exploit the property of $\mathbf{w}^*$ to derive Eq. (13), which serves as an indirect optimizer for maximizing the $\text{NDCG}(\boldsymbol{D})@N$ by reweighting the prior weights of the GMM. Intuitively, the vector $\mathbf{w}^*$ derived in line 11 plays a role to select mixture components in the GMM.

In the testing phase, each song is encoded into a BOF vector by the GMM with the relearned prior weights using Eq. (3). Then, the audio-based similarity between any two songs s_i and s_j is computed as the inner product of $\mathbf{x}_i$ and $\mathbf{x}_j$, without the need to apply any stacking transformation in the BOF space. In the experiments, the proposed method implemented in this way is denoted as DiagGMM.

Algorithm 1. The learning algorithm

Input: Initial GMM parameters $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$, k=1,…,K;
A tagged music corpus $\boldsymbol{D}$: a set of frames $\mathbf{V}_i$ for s_i, i=1,…,N, and tag similarity matrix S_Y from Eq. (5);

Output: Learned GMM prior $\{\hat{\pi}_k\}$;

01: Initialize $\pi_k^{(0)}$ to be $1/K$;
02: Iteration index $t \leftarrow 0$;
03: $L(t) \leftarrow 0$;
04: **while** $t \geq 0$ **do**
05: Encode $\mathbf{V}_i$ into $\mathbf{x}_i$ with Eq. (3) using $\{\pi_k^{(t)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$;
06: Compute S_X with Eq. (4);
07: $t \leftarrow t + 1$;
08: $L(t) \leftarrow$ NDCG($\boldsymbol{D}$)@N with Eq. (7) using S_X and S_Y;
09: **If** $(L(t)-L(t\text{-}1))/L(t) < 0$ **then**
10: Return $\hat{\pi}_k \leftarrow \pi_k^{(t-1)}$ and **break**;
11: Compute $\mathbf{w}^*$ with Eq. (12) using S_X and S_Y;
12: **for** k=1,…,K, **do**

13: $$\pi_k^{(t)} \leftarrow \frac{w_k \pi_k^{(t-1)}}{\sum_{q=1}^{K} w_q \pi_q^{(t-1)}}; \qquad (13)$$

(where w_k is the k-th element in $\mathbf{w}^*$)
14: **end for**
15: **end while**

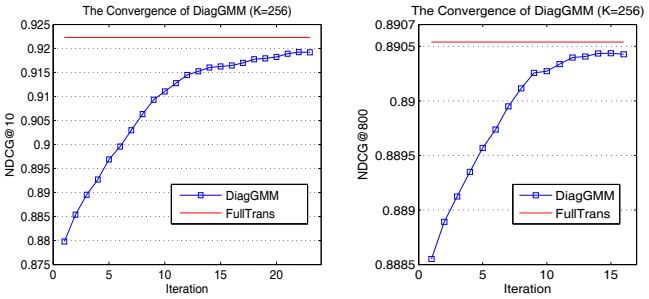
4. EVALUATIONS

4.1 Datasets

We evaluate the proposed method on the MajorMiner and Magnatagatune datasets in a query-by-example MIR scenario. Both datasets are generated from social tagging games with a purpose (GWAP) [11, 12] to collect reliable and useful tag labels. The MajorMiner dataset has been a well-known benchmark in MIREX since 2008. The one used in this paper is crawled from the MajorMiner website in March 2011. It contains 2,472 10-second music clips and 1,031 raw tags. After exacting the high frequency tags and merging the redundant tags, 76 tags are left. The Magnatagatune dataset [12], which contains 25,860 30-second audio clips and 188 pre-processed tags, is downloaded from [21]. To construct $\boldsymbol{F}$, we randomly select 25% and 2% of frames from the two datasets, respectively. For MajorMiner, $\boldsymbol{F}$ contains 235,000 frames, while for Magnatagatune, $\boldsymbol{F}$ contains 535,800 frames. The $\boldsymbol{F}$ constructed in this way is blind to song-level information. To prevent bias in the tag-based similarity computation of S_Y, we ignore the clips labeled with fewer tags. For the MajorMiner dataset, 1,200 clips having at least 5 tags are left. For the Magnatagatune dataset, 3,764 clips having at least 7 tags are left.

4.2 Experimental Results and Discussions

In the experiments, we repeat three-fold cross-validation 10 times on the MajorMiner dataset, which is divided into three folds at random. In each run, two folds are used for training the transformation matrix of the FullTrans and DiagTrans methods or relearning the prior weights of the GMM for the DiagGMM method, while the remaining fold, which *serves as both the test queries and the target database to retrieve*, is used for the *leave-one-out* audio-based MIR outside test. For the Magnatagatune dataset, all clips have been divided into 16 folds to prevent that two or more clips originated from the same song occur in different folds. We merge the 16 folds into 4 folds and perform four-fold cross-validation. The NDCG@P in Eq. (7) is used as the evaluation metric in both inside and outside tests.



(a) (b)

Figure 1. The learning curve in terms of (a) NDCG@10 and (b) NDCG@800 evaluated on the MajorMiner training data.

First, we examine the learning process of DiagGMM on the MajorMiner dataset. Figure 1 shows an example learning curve in terms of NDCG for one of the three-fold cross-validation runs. The curve is equivalent to the *inside test* performance evaluated on the training data. We can see that the learning curve of DiagGMM (K=256) increases monotonically till convergence, although DiagGMM can only improve the NDCG of the training data indirectly as discussed in Section 3.3. DiagGMM gains an absolute increase of 0.04 in NDCG@10 and 0.002 in NDCG@800. The NDCG of FullTrans can be considered an upper bound for DiagGMM since it adopts a direct optimization strategy.

Next, we evaluate OrigGMM and the VQ-based method on the MajorMiner dataset. There is no need to divide the data into three folds since no supervised learning is involved in the methods. From the MIR results shown in Table 2, we observe that replacing the priors of the GMM trained from $\boldsymbol{F}$ with a uniform distribution enhances the performance. We also observe that, even with a large K, OrigGMM outperforms VQ-based BOF modeling. The results demonstrate the better modeling ability of the GMM over the K-means derived codebook.

Finally, we compare DiagGMM with three baselines, i.e., FullTrans, DiagTrans, and OrigGMM. The results of three-fold cross-validation on the MajorMiner dataset are shown in Figure 2, while the results of four-fold cross-validation on the Magnatagatune dataset are shown in Figure 3. From Figures 2 and 3, it is obvious that the proposed DiagGMM outperforms all other methods in most cases. The conventional BOF approach does face a glass ceiling when K is

too large, as evidenced by the observation that the performance of OrigGMM saturates at around K=1,024 for MajorMiner (10-second clips) and K=2,048 for Magnatagatune (30-second clips). The proposed DiagGMM enhances the performance over the glass ceiling of OrigGMM with a smaller K, e.g., DiagGMM with K=512 outperforms OrigGMM with K=2,048 on the MajorMiner dataset. FullTrans outperforms DiagTrans and DiagGMM only when K is small. However, FullTrans tends to saturate early since it has more parameters to train and thus requires more training data, compared with DiagTrans and DiagGMM. In Figure 1, the performance of FullTrans shows an upper bound of DiagGMM in inside test; however, in outside test, DiagGMM outperforms FullTrans except when K is small. The experimental results in Figures 2 and 3 demonstrate the excellent generalization ability of DiagGMM, which learns the similarity of audio music by relearning the priors of the GMM instead of a transformation in the BOF vector space.

5. CONCLUSIONS

In this paper, we have addressed a novel research direction that the audio-based music similarity computation can be learned by minimizing the similarity gap or maximizing the NDCG measure with respect to the parameters of the encoding reference in BOF representation. We have implemented the idea by learning the prior weights of the GMM from tagged music data. The experimental results demonstrate the effectiveness of the proposed method, which gives a potential to guide MIR systems that employ BOF representation, e.g., the DiagGMM can be directly combined with the codeword Bernoulli average (CBA) method [13], a well-known automatic music tagging method.

NDCG	@5	@10	@20	@30
OrigGMM (K=2,048) w/o Prior	0.9382	0.9015	0.8753	0.8674
OrigGMM (K=2,048) w Prior	0.9322	0.8992	0.8743	0.8669
VQ-based (K=2,048) Histogram	0.9297	0.8930	0.8721	0.8650

Table 2. The results of OrigGMM and the VQ-based method on the complete MajorMiner dataset (1,200 clips).

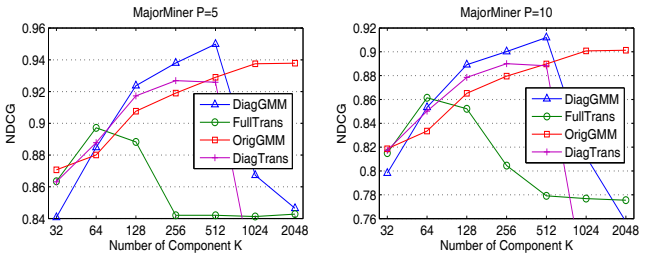


Figure 2. The results in terms of NDCG@5 and NDCG@10 on the MajorMiner dataset with different K.

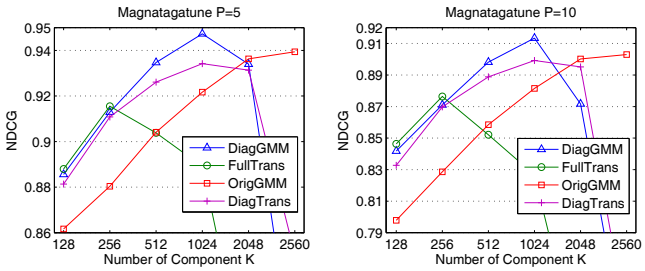


Figure 3. The results in terms of NDCG@5 and NDCG@10 on the Magnatagatune dataset with different K.

6. REFERENCES

[1] M. Slaney, K. Weinberger, and W. White: "Learning a metric for music similarity," *ISMIR*, 2008.

[2] J.-J. Aucouturier and F. Pachet: "Music similarity measures: What's the use?," *ISMIR*, 2002.

[3] M. Mandel and D. Ellis: "Song-level features and SVMs for music classification," *ISMIR*, 2005.

[4] E. Pampalk: "Audio-based music similarity and retrieval: Combining a spectral similarity model with information extracted from fluctuation patterns," *ISMIR*, 2006.

[5] M. Hoffman, D. Blei and P. Cook: "Content-based musical similarity computation using the hierarchical Dirichlet process," *ISMIR*, 2008.

[6] K. West and P. Lamere: "A model-based approach to constructing music similarity functions," *EURASIP Journal on Advances in Signal Processing*, 2007(1), 1–10, 2007.

[7] L. Barrington, A. Chan, D. Turnbull, and G. Lanckriet: "Audio information retrieval using semantic similarity," *ICASSP*, 2007.

[8] M. Levy and M. Sandler: "Music information retrieval using social tags and audio," *IEEE TMM*, 11(3), 383-395, 2009.

[9] J. H. Kim, B. Tomasik, and D. Turnbull: "Using artist similarity to propagate semantic information," *ISMIR*, 2009.

[10] B. McFee, L. Barrington and G. Lanckriet: "Learning similarity from collaborative filters," *ISMIR*, 2010.

[11] M. Mandel and D. Ellis: "A web-based game for collecting music metadata," *J. New Mus. Res.*, 37(2), 151–165, 2008.

[12] E. Law and L. von Ahn: "Input-agreement: A new mechanism for data collection using human computation games," *ACM CHI*, 2009.

[13] M. Hoffman, D. Blei and P. Cook: "Easy as CBA: A simple probabilistic model for tagging music," *ISMIR*, 2009.

[14] O. Lartillot and P. Toiviainen: "A Matlab toolbox for musical feature extraction from audio," *DAFx*, 2007.

[15] G. Marques, et al.: "Additional evidence that common low-level features of individual audio frames are not representative of music genre," *SMC Conference*, 2010.

[16] J.-J. Aucouturier, B. Defreville and F. Pachet: "The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music," *J. Acoust. Soc. Am.*, 122(2), 881–91, 2007.

[17] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno: "An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model," *IEEE TASLP*, 16(2), 435–447, 2008.

[18] K. Jarvelin and J. Kekalainen: "Cumulated gain-based evaluation of IR techniques," *ACM Trans. on Information Systems*, 20(4), 422–446, 2002.

[19] Y. Zhang and Z.-H. Zhou: "Multi-label dimensionality reduction via dependence maximization," *AAAI*, 2008.

[20] J. V. Davis, B. Kulis, P. Jain, S. S. and I. S. Dhillon: "Information-theoretic metric learning," *ICML*, 2007.

[21] http://tagatune.org/Magnatagatune.html