MUSIC TAG ANNOTATION AND CLUSTERING USING LATENT MUSIC SEMANTIC ANALYSIS

Ju-Chiang Wang[1,2]*, Meng-Sung Wu*[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
E-mail: {asriver, wums, whm}@iis.sinica.edu.tw, skjeng@cc.ee.ntu.edu.tw

ABSTRACT

Music tags include different types of musical information. The tags of same or different types can be assigned together by human to a specific song. This may lead to some specific tag co-occurrence patterns among auditorily similar songs. In this paper, we propose a novel generative approach via Latent Music Semantic Analysis (LMSA) to model and predict the tag co-occurrence pattern of a song. The LMSA-based approach jointly models two types of features, namely, auditory music features and tag-based text features. We employ a Gaussian mixture model (GMM) or a codebook to represent the auditory feature references and a tag-based music semantic model to model the tag co-occurrence patterns given the GMM-based or vector quantized auditory feature representation. We demonstrate the capability of the LMSA-based approach in music semantic exploration and music tag clustering. In addition, the results of music tag annotation experiments show that our method outperforms the baseline Codeword Bernoulli Average (CBA) method.

1. INTRODUCTION

In recent years, music tagging has generated a great deal of interest among researchers in the field of music information retrieval (MIR) [1]. For example, Turnbull et al. [2] model the feature distribution of each tag with a Gaussian mixture model (GMM) and estimate the model's parameters with a weighted mixture hierarchies expectation maximization algorithm. In contrast to using probability models, Eck et al. [3] use AdaBoost to automatically generate audio tags for music recommendation. In addition, Hoffman et al. [4] propose the codeword Bernoulli average (CBA) model, which applies Bernoulli distribution to model the probability between a tag and a codeword based on each song's vector-quantized (VQ) histogram in a music corpus. These approaches model and predict each unique tag independently without considering the counts of tags assigned to a song and the tag co-occurrence phenomenon of a song. Several recent studies [5, 6] consider the correlation between any two tags for music tag annotation. They employ a two-stage classification, in which a stacked classifier is used to combine the outputs of the individual tag classifiers, to improve the performance with higher computational complexity.

Music tags are a natural way to describe the general musical concepts since people tend to mentally tag a piece of music with specific words when they listen to it. The tags can include different types of musical information, such as genre, mood, instrumentation, personal preference, original artists, and particular usages. The tags of same or different types can be assigned together by human to a specific song. This may lead to several specific tag co-occurrence (denoted as co-tag hereafter) patterns among auditorily similar songs. For example, some instrumental or timbre tags are inspired directly by auditory cues, such as guitar, drum, rap, saxophone, piano, synth, and drum-machine. These instrumental tags usually result in a series of consequent tags, e.g., electric guitar, distortion, and drum commonly result in rock, loud, and punk; saxophone and piano usually lead to jazz and soft; rap mostly co-occurs with hip-hop; synth and drum-machine often give electronic and techno tags. The tags co-occur frequently in many songs of a dataset are regarded as having strong relations among them [7]. As a result, we are interested in investigating the co-tag patterns which may imply some specific musical aspects. In this paper, we explore and discover the music co-tag patterns through codebook/ GMM learning on auditory features and *Latent Music Semantic Analysis* (LMSA) on tag labels with counts.

In this paper, we propose a novel probabilistic generative model via *Latent Music Semantic Analysis* to model and predict the tag co-occurrence pattern of a song. We assume that there are several latent co-tag patterns in human minds. When tagging a song, people commonly choose one or more un-describable co-tag patterns according to the auditory musical characteristics of the song. Although we cannot describe exactly what the latent co-tag patterns and the auditory musical characteristics are, we believe that there is a strong linkage between them. Consequently, as shown in Figure 1, we introduce a hidden layer of latent feature classes in the co-tag generative flow to link the latent co-tag patterns and the music features.

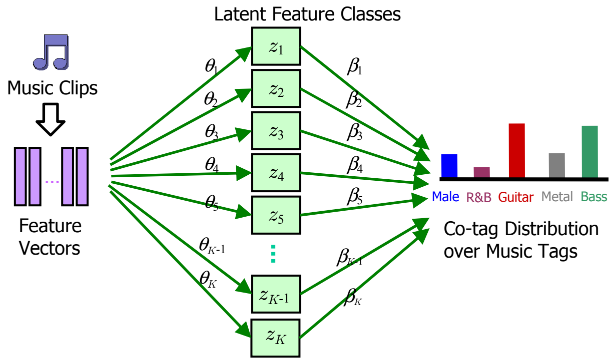

Figure 1. The overall co-tag generative flow.

Assume that there are K latent feature classes z_k, k=1,…,K, and, for each class z_k, the latent co-tag pattern is modeled by β_k. A song is first transformed into a sequence of feature vectors, after which the posterior weight (denoted as θ_k) of a certain z_k given the song is generated by a pre-trained model. Theoretically, with a large K, all seen co-tag patterns can be generated approximately by the convex combination of β_k and θ_k, k =1,…,K. With β_k, k =1,…,K, we can predict the co-tag pattern for an untagged song based on its θ_k, k =1,…, K. If the song's audio features can be completely described by a certain latent feature class z_k, i.e., θ_k =1, and θ_i=0 for all $i \neq k$, then its co-tag pattern would exactly follow the pattern β_k. With β_k, k =1,…,K, we can also achieve tag clustering by assigning the highly co-occurred tags into a cluster.

To implement the idea, we assume that each latent co-tag pattern can be modeled by a multinomial distribution, and the latent feature classes can be described by a vector codeword of a codebook or a mixture component of a Gaussian mixture model (GMM). Then, all existing co-tag patterns can be generated by a mixture of the multinomial models, i.e., a mixture of latent co-tag patterns.

The remainder of this paper is organized as follows. In Section 2, we describe music feature extraction and representation. In Section 3, we introduce the latent music semantic analysis and explain how to apply it in music tag annotation and clustering. The evaluations and results are detailed in Section 4. Finally, we summarize our conclusions and discuss our future work in Section 5.

2. AUDIO FEATURE EXTRACTION AND SONG-LEVEL REPRESENTATION

In this section, we describe the auditory music features used in this work, and explain how we convert the frame-based feature vectors of a song into a fixed-dimensional vector representation through a music feature reference, namely, a codebook or a GMM.

2.1. Audio Feature Extraction

We use MIRToolbox 1.3 for music audio feature extraction [8]. As shown in Table 1, we consider four types of features in this work, including dynamic, spectral, timbre, and tonal features. To ensure the alignment and prevent the mismatch of different features in a vector, all the features are extracted from the same fixed-size short-time frame. Given a song, a sequence of 70-dimensional feature vectors is extracted with a 50ms frame size and half-shifting.

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

Category	Feature Description	Dim
dynamics	rms	1
spectral	centroid	1
	spread	1
	skewness	1
	kurtosis	1
	entropy	1
	flatness	1
	rolloff at 85%	1
	rolloff at 95%	1
	brightness	1
	roughness	1
	irregularity	1
timbre	zero crossing rate	1
	spectral flux	1
	MFCC	13
	delta MFCC	13
	delta-delta MFCC	13
tonal	key clarity	1
	key mode possibility	1
	HCDF	1
	chroma peak	1
	chroma centroid	1
	chroma	12

2.2. Song-level Feature Representation

To train the auditory feature reference, we first normalize the 70-dimensional frame-based feature vectors in each dimension to mean 0 and standard deviation 1. Then, we define a set of "latent feature classes" represented by z_k, k=1,…,K, and each corresponds to the k-th codeword $\mathbf{c}_k$ in the codebook, or the k-th Gaussian component with mixture weight π_k, mean vector $\boldsymbol{\mu}_k$, and covariance matrix $\boldsymbol{\Sigma}_k$ in the GMM. The codebook is trained by the K-means algorithm, and the GMM is fitted with the EM algorithm.

With the codebook, a song s_n can be represented as a fixed-dimensional vector of codeword histogram, which is constructed by applying vector quantization to its each frame-based feature vector $\mathbf{x}_{nt}$:

$$VQ(\mathbf{x}_{nt}; \boldsymbol{C}) = \arg\min_k \left\| \mathbf{x}_{nt} - \mathbf{c}_k \right\|, \tag{1}$$

where $\boldsymbol{C}$ is the pre-trained codebook and $\|\ .\ \|$ is the

Euclidean distance. By assuming that each frame contributes equally to the song, we obtain the VQ-based histogram $\boldsymbol{\theta}_n$ whose k-th component θ_{nk} is computed by

$$\theta_{nk} = p(z_k \mid s_n; \boldsymbol{C}) = \frac{1}{T_n}\sum_{t=1}^{T_n} 1\{VQ(\mathbf{x}_{nt}; \boldsymbol{C}) = k\}, \qquad (2)$$

where T_n is the number of frames in song s_n and the $1\{a=b\}$ function returns 1 if a equals to b. Then each element in $\boldsymbol{\theta}_n$ is normalized to sum to 1.

With the GMM, the posterior probability of z_k given music feature vector $\mathbf{x}_{nt}$ is computed by

$$p(z_k \mid \mathbf{x}_{nt}; \boldsymbol{\Lambda}) = \frac{p(z_k; \pi_k)\, p(\mathbf{x}_{nt} \mid z_k; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} p(z_l; \pi_l)\, p(\mathbf{x}_{nt} \mid z_l; \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}, \qquad (3)$$

where $\boldsymbol{\Lambda}$ is the parameter set of the pre-trained GMM. The k-th component θ_{nk} of the GMM-based posterior representation $\boldsymbol{\theta}_n$ is computed by

$$\theta_{nk} = p(z_k \mid s_n; \boldsymbol{\Lambda}) = \frac{1}{T_n}\sum_{t=1}^{T_n} p(z_k \mid \mathbf{x}_{nt}; \boldsymbol{\Lambda}). \qquad (4)$$

The two song-level feature representations of a song can be modeled by the generative process of a tag-based music semantic model, as will be described later.

3. LATENT MUSIC SEMANTIC ANALYSIS

We are motivated by the probabilistic latent aspect model, which has been widely used in text document modeling [9], in tag-based latent music semantic analysis. By treating the music tag labels as the text features of a song and representing them by a "bag-of-tags" vector, the tag labels of a music corpus can be modeled by a set of latent multinomial distributions. Suppose we have a music corpus with N songs, each denoted by s_n, n=1,…,N; and let each song's tag count $c(n,m)$, m=1,…,M, be an non-negative integer representing the number of times that tag w_m has been assigned to song s_n. The co-tag over the predefined M tags are denoted as $\mathbf{w} = (w_1, w_2, \ldots, w_M)$, and $p(\mathbf{w} \mid s_n; \mathbf{B})$ represents the tag-based latent music semantic model with parameter set $\mathbf{B}$.

3.1. The Generative Process

There are three steps to generate the co-tag pattern $\mathbf{w}$ of song s_n. First, a latent feature class z_k is chosen with the probability θ_{nk}:

$$p(z_k \mid s_n; \boldsymbol{\theta}_n) = \theta_{nk}. \qquad (5)$$

The probability of z_k can be viewed as a mixture prior that has been determined in the auditory feature representation stage. The prior plays a constraint role of the auditory features in the Bayesian learning framework. Second, a tag w_m of song s_n can be generated by the marginal distribution over all latent feature class z_k, k =1,…,K:

$$p(w_m \mid s_n; \boldsymbol{\theta}_n, \mathbf{B}) = \sum_{k=1}^{K} \theta_{nk}\beta_{km}, \qquad (6)$$

where β_{km} represents the probability of w_m in the k-th latent co-tag pattern. Therefore, the co-tag $\mathbf{w}$ of s_n can be generated by the multinomial distribution as expressed by

$$p(\mathbf{w} \mid s_n; \boldsymbol{\theta}_n, \mathbf{B}) = \prod_{m=1}^{M} p(w_m \mid s_n; \boldsymbol{\theta}_n, \mathbf{B})^{c(n,m)} = \prod_{m=1}^{M}\left(\sum_{k=1}^{K}\theta_{nk}\beta_{km}\right)^{c(n,m)}. \qquad (7)$$

Given the music corpus with $\boldsymbol{\theta}_n$ and $c(n,m)$, m=1,…,M, n=1,…,N, the full log-likelihood function is

$$\begin{aligned} L &= \log p(\mathbf{w}; \boldsymbol{\Theta}, \mathbf{B}) = \sum_{n=1}^{N} \log\left[p(s_n) + p(\mathbf{w} \mid s_n; \boldsymbol{\theta}_n, \mathbf{B})\right] \\ &= \sum_{n=1}^{N}\left[\log p(s_n) + \sum_{m=1}^{M} c(n,m) \log \sum_{k=1}^{K} \theta_{nk}\beta_{km}\right], \end{aligned} \qquad (8)$$

where $\boldsymbol{\Theta}$ represents $\{\boldsymbol{\theta}_n\}$, n=1,…,N, and $p(s_n)$ is assumed to be uniformly distributed and can be ignored in the following learning procedure.

3.2. Model Inference with the EM Algorithm

The model expressed in Eq. (7) can be fitted with respect to $\mathbf{B}$ and $\boldsymbol{\Theta}$ with maximum-likelihood (ML) estimation. However, in the generative process, $\boldsymbol{\Theta}$ has been determined in the song-level feature representation stage in Eq. (2) for the VQ-based representation or in Eq. (4) for the GMM-based representation. Therefore, we only need to estimate $\mathbf{B}$. Given the song-level feature representation $\boldsymbol{\theta}_n$ and tag counts $c(n,m)$ of song s_n, we apply the EM algorithm to maximize Eq. (8) with respect to $\mathbf{B}$ in the presence of latent variable z.

In the E-step, the posterior probability of z_k given song s_n and tag w_m is

$$p(z_k \mid s_n, w_m; \boldsymbol{\theta}_n, \boldsymbol{\beta}_k) = \frac{\theta_{nk}\beta_{km}}{\sum_{q=1}^{K}\theta_{nq}\beta_{qm}}. \qquad (9)$$

In the M-step, $\mathbf{B}$ is updated based on the expected complete data log-likelihood over the posterior probabilities computed in the E-step. The update rule for β_{km} is

$$\beta_{km} \leftarrow p(w_m \mid z_k; \boldsymbol{\Theta}, \boldsymbol{\beta}_k) = \frac{\sum_{n=1}^{N} c(n,m)\, p(z_k \mid s_n, w_m; \boldsymbol{\theta}_n, \boldsymbol{\beta}_k)}{\sum_{n=1}^{N}\sum_{r=1}^{M} c(n,r)\, p(z_k \mid s_n, w_r; \boldsymbol{\theta}_n, \boldsymbol{\beta}_k)}. \qquad (10)$$

where $\boldsymbol{\beta}_k$ is the parameter set (probability vector) of the k-th latent multinomial distribution $\{\beta_{km}\}$, m=1,…,M, in $\mathbf{B}$ and represents the latent co-tag pattern of the k-th latent feature class and *gives a semantic meaning to the latent feature class*. Therefore, the training process is a kind of "music semantic analysis". We can apply the model in tag annotation, i.e., predicting tags for a new song, or tag clustering, i.e., clustering tags in a labeled music corpus.

3.3. Music Tag Annotation and Clustering

For tag annotation, an untagged song s is first transformed into the song-level feature representation $\boldsymbol{\theta}$. Then, the affinity score of tag w_m for song s is computed by the convex combination among mixture probabilities, each with parameter $\boldsymbol{\beta}_k$:

$$p(w_m \mid s;\boldsymbol{\theta},\mathbf{B}) = \sum_{k=1}^{K} p(z_k \mid s;\boldsymbol{\theta}) p(w_m | z_k;\boldsymbol{\beta}_k) = \sum_{k=1}^{K} \theta_k \beta_{km}. \quad (11)$$

The computational cost of our music tag annotation method only depends on the number of latent feature classes K and the number of the songs N in the training corpus. Since our model considers all the tags jointly and all parameters in $\mathbf{B}$ are learned at once, it is more efficient in training duration than the existing approaches that apply an independent classifier for each tag.

For tag clustering, we are interested in which tags in a latent co-tag pattern contribute more to the corresponding latent feature class. We simply assign the tag w_m to cluster k^* that has the largest β_{km} among all latent feature classes z_k, k=1,…,K, as follows,

$$\text{Cluster}(w_m) = \arg\max_k \beta_{km}, \quad k = 1,...,K. \quad (12)$$

It may happen that no tag is assigned to a certain cluster h because the latent co-tag pattern $\boldsymbol{\beta}_h$ of the cluster might be uniformly distributed and no tag is informative for the cluster. Moreover, it is inevitable that there will be more empty clusters if the number of latent feature classes K is larger than the number of distinct tags M. The tags belonging to the same cluster tend to jointly have large probabilities; this can be the evidence of the strong relations among them.

4. EVALUATION

We evaluate the proposed approach on the MajorMiner dataset. The MajorMiner website employs a game to gather reliable text labels for music [10]. Each player labels randomly given music clips (each about 10 seconds long) by listening to them without any meta-information. If two players assign the same text label to a particular music clip, the label is adopted by the system. Hence, each music clip's tag count is at least 2. We download all the music clips associated with the most commonly used 45 tags from the MajorMiner website. The resulting dataset contains 2,742 audio clips. In the dataset, the count of a tag given to a music clip is at most 12.

To train the codebook and GMM mentioned in Sec. 2, we randomly select 25% of the frame vectors in the dataset, which yields approximate 235,000 vectors. The codebook, GMM and tag-based latent music semantic model are trained by using the MATLAB software with the stopping criterion that the objective function is reduced (for the codebook) or increased (for the GMM and latent music semantic model) by less than 0.0001.

4.1. Music Tag Clustering

In the music tag clustering experiments, we use the GMM-based auditory feature representation for building $\boldsymbol{\theta}$. We demonstrate the musical aspect with the top 6 tags in a co-tag pattern when K =16 in Table 2. It seems that the 6 tags in a column match some specific musical aspect. For example, the first music aspect pictures some soft songs leaded by female singers, and accompanied with piano, guitar and strings; while the tags in the sixteenth musical aspect are definitely the rock' n' roll stuff. However, some tags seem to be redundant because of their high frequencies in the MajorMiner dataset. The redundancy would be reduced if we have a larger set of tags or a larger set of balanced tagged music corpus. The results of tag clustering when K =16 are shown in Table 3. Each of the 45 tags is clustered to one of the 16 clusters. Generally speaking, the clustering results pretty match some musical common senses. We have also performed tag clustering on the CAL500 dataset [2], which contains 500 of western popular songs. Each clip has been manually labeled by at least three humans following 174 pre-defined text labels. We select a subset of 111 tags, and the tag clustering result of CAL-500 is shown in Table 4.

4.2. Music Tag Annotation

In the music tag annotation experiments, the number of latent feature classes is set in between K=64 and K=2,048. We repeat three-fold cross-evaluation 10 times, i.e., in a set of randomly split three folds, 1648 clips are used for training and 824 clips for testing. Four systems are compared, including CBA, GBA, cwLMSA, and postLMSA. The CBA system [4] is the baseline. We modify the CBA method to the GBA (Gaussian Bernoulli Average) method replacing the codebook into a GMM. The cwLMSA and postLMSA systems employ the proposed VQ-based and GMM-posterior-based LMSA approaches, respectively. In our implementation, each system annotates the top *five* tags with the highest affinity scores to a clip. The averaged performance, as shown in Figure 2, is evaluated in terms of the F-measure and the area under the ROC curve per clip (AUC Per Clip).

The results demonstrate that the proposed postLMSA approaches outperform the baseline CBA approach in terms of all evaluation metrics. The tag-based semantic model can better generalize the music tag modeling than independent Bernoulli models. Since K represents the resolution of the latent feature classes, the performance increases as K increases. We observe that the GMM shows significantly better ability in audio feature modeling over the codebook when K is small, as shown in the comparisons between CBA vs. GBA, and cwLMSA vs. postLMSA, respectively. However, when K is large, the advantage of GMM becomes small.

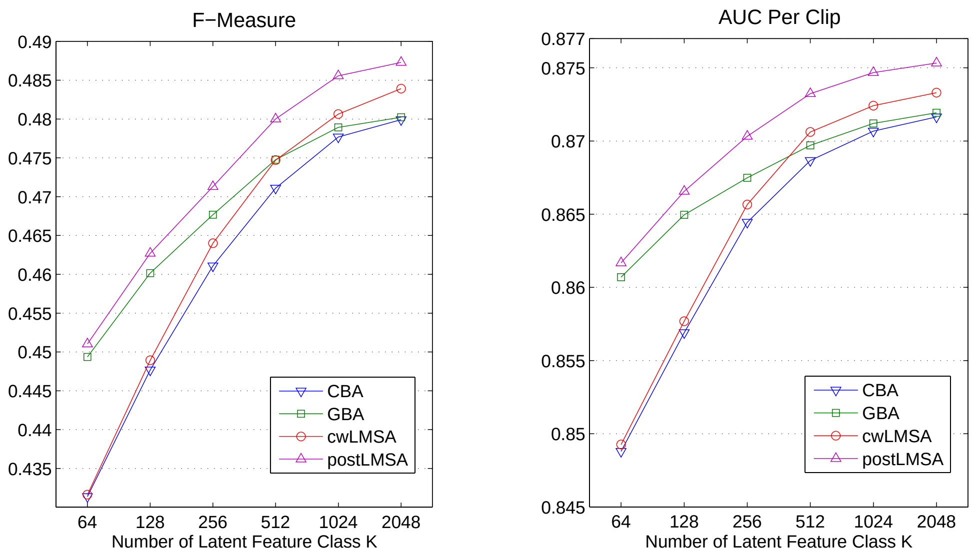

Figure 2. The results of music tag annotation.

5. CONCLUSION

We have proposed a novel LMSA-based music tagging approach that jointly models two types of features, namely, auditory music features and tag-based text features. We have demonstrated its capability in tag-based music aspect exploration and music tag clustering. In addition, it outperforms the CBA approach with lower computational complexity in training. The latent music semantic analysis and tag clustering technique can be a potential solution for advanced music information retrieval and exploration.

In our implementation, we employ a GMM or a codebook to model the auditory features and a tag-based music semantic model to model the co-tag patterns given the GMM-based or VQ-based music feature representation. This implementation adopts a two-stage optimization, i.e., it first optimizes the auditory feature model, and then optimizes the co-tag model based on the fixed auditory feature model. In our future work, we will try to jointly optimize the two models under a common latent condition. We will also investigate other probabilistic models that have more layers of generative processes and hidden conditions.

6. ACAKNOWLEDGEMENTS

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.

7. REFERENCES

[1] Paul Lamere, "Social tagging and music information retrieval," *Journal of New Music Research*, 37(2), pp. 101–114, 2008.

[2] D. Turnbull, L. Barrington, D. Torres, and G. R. G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *IEEE Transactions on Audio, Speech and Language Processing*, 16(2), pp. 467–476, 2008.

[3] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green, "Automatic generation of social tags for music recommendation", *Advances in Neural Information Processing System (NIPS) 20*, MIT Pressed, 2008.

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

[5] S. R. Ness, A. Theocharis, G. Tzanetakis, and L. G. Martins, "Improving automatic music tag annotation using stacked generalization of probabilistic SVM outputs," *Proc. of ACM MM*, 2009.

[6] R. Miotto, L. Barrington and G. R. G. Lanckriet, "Improving auto-tagging by modeling semantic co-occurrences," *Proc. of ISMIR*, 2010.

[7] G. Begelman, P. Keller, F. Smadja, "Automated tag clustering: Improving search and exploration in the tag space," *Proc. of WWW*, 2006.

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

[9] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning*, 42(1), pp. 177-196, 2001.

[10] M. Mandel and D. P. W. Ellis: "A web-based game for collecting music metadata," *Journal of New Music Research*, 37(2), pp. 151–165, 2008.

Table 2. The latent co-tag patterns (musical aspects) described by the top 6 tags when *K* =16.

1	piano	guitar	slow	female	strings	vocal
2	electronic	synth	dance	drums	beat	techno
3	male	synth	drums	guitar	vocal	female
4	drums	electronic	female	dance	techno	synth
5	guitar	synth	drums	electronic	bass	punk
6	synth	electronic	drums	male	bass	techno
7	synth	pop	drums	male	dance	guitar
8	quiet	ambient	synth	electronic	guitar	noise
9	synth	guitar	drums	bass	electronic	slow
10	guitar	rock	drums	pop	male	bass
11	guitar	male	drums	synth	country	rock
12	piano	jazz	synth	electronic	ambient	quiet
13	rap	hip-hop	male	funk	female	beat
14	saxophone	synth	electronic	drums	jazz	guitar
15	jazz	saxophone	female	piano	trumpet	vocal
16	rock	guitar	drums	male	vocal	punk

Table 3. The results of tag clustering when *K* =16 (16 clusters) for MajorMiner.

1	2	3	4	5	6	7	8
slow strings organ	dance beat drum-machine	vocal voice acoustic	drums techno house horns	guitar bass punk metal	electronic electronica	fast	ambient quiet noise
9	**10**	**11**	**12**	**13**	**14**	**15**	**16**
synth soft r&b keyboard	pop	male country	piano instrumental	hip-hop rap funk	80s british solo	female jazz saxophone trumpet	rock distortion loud

Table 4. The results of tag clustering when *K* =16 (16 clusters) for CAL-500. Each tag is separated by a comma.

1	Genre--_Country_Blues, Genre--_Gospel, Genre--_Soul, Genre--_Swing, Instrument_-_Female_Lead_Vocals, Instrument_-_Harmonica, Instrument_-_Saxophone, Instrument_-_Trombone, Instrument_-_Trumpet, Usage-With_the_family, Vocals-Gravelly, Vocals-Spoken, Vocals-Strong, Genre-Best-Blues, Instrument_-_Trumpet-Solo
2	Genre--_Dance_Pop, Instrument_-_Drum_Machine, Instrument_-_Sequencer, Instrument_-_Synthesizer, Usage-At_a_party, Usage-Getting_ready_to_go_out, Usage-Waking_up, Vocals-Altered_with_Effects, Instrument_-_Female_Lead_Vocals-Solo
3	Genre--_Bebop, Instrument_-_Tambourine, Genre--_Contemporary_Blues, Usage-At_work, Genre-Blues, Usage-Reading, Instrument_-_Ambient_Sounds, Usage-Romancing, Instrument_-_Hand_Drums, Vocals-Breathy, Instrument_-_Organ, Instrument_-_Harmonica-Solo
4	Genre--_Alternative, Vocals-Duet, Vocals-Falsetto, Vocals-High-pitched, Genre-Best--_Alternative, Instrument_-_Electric_Guitar_(distorted)-Solo
5	Genre--_Alternative_Folk, Genre-Electronica, Genre-Best-Electronica, Instrument_-_Male_Lead_Vocals-Solo
6	Genre--_Cool_Jazz, Usage-Studying, Genre-Jazz, Instrument_-_Piano, Genre-Best-Jazz, Instrument_-_Violin/Fiddle, Instrument_-_Piano-Solo, Usage-Going_to_sleep, Instrument_-_Saxophone-Solo, Usage-Sleeping
7	Genre-Country, Instrument_-_Acoustic_Guitar, Instrument_-_Horn_Section, Genre-Best-Folk, Instrument_-_Acoustic_Guitar-Solo
8	Genre--_Soft_Rock, Vocals-Low-pitched, Genre-Bluegrass, Vocals-Vocal_Harmonies, Genre-Folk, Genre-Best--_Soft_Rock, Genre-R&B, Genre-Best-R&B, Instrument_-_Backing_vocals, Instrument_-_String_Ensemble
9	Genre-World, Usage-Exercising, Usage-Intensely_Listening, Vocals-Aggressive, Vocals-Call_&_Response, Genre-Best-World
10	Genre--_Funk, Genre-Hip_Hop/Rap, Vocals-Rapping, Genre-Best-Hip_Hop/Rap
11	Genre--_Electric_Blues, Genre--_Roots_Rock, Genre-Rock, Instrument_-_Bass, Instrument_-_Drum_Set, Instrument_-_Electric_Guitar_(clean)-Solo
12	Genre--_Contemporary_R&B, Instrument_-_Samples, Usage-Driving, Vocals-Monotone, Genre-Best-Pop
13	Genre--_Brit_Pop, Genre--_Classic_Rock, Genre-Pop, Instrument_-_Male_Lead_Vocals, Vocals-Screaming, Genre-Best--_Metal/Hard_Rock
14	Genre--_Metal/Hard_Rock, Instrument_-_Electric_Guitar_(clean), Instrument_-_Electric_Guitar_(distorted)
15	Genre--_Punk, Genre-Best--_Classic_Rock, Genre-Best-Rock, Genre-Best--_Punk
16	Genre--_Singer_/_Songwriter, Usage-Cleaning_the_house, Usage-Hanging_with_friends, Vocals-Emotional, Genre-Best--_Soul, Genre-Best-Country