Multilingual I-Vector based Statistical Modeling for Music Genre Classification

Jia Dai, Wei Xue, Wenju Liu

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China
University of Chinese Academy of Sciences, Beijing, China
{jia.dai, wxue, lwj}@nlpr.ia.ac.cn

Abstract

For music signal processing, compared with the strategy which models each short-time frame independently, when the long-time features are considered, the time-series characteristics of the music signal can be better presented. As a typical kind of long-time modeling strategy, the identification vector (i-vector) uses statistical modeling to model the audio signal in the segment level. It can better capture the important elements of the music signal, and these important elements may benefit to the classification of music signal. In this paper, the i-vector based statistical feature for music genre classification is explored. In addition to learn enough important elements for music signal, a new multilingual i-vector feature is proposed based on the multilingual model. The experimental results show that the multilingual i-vector based models can achieve better classification performances than conventional short-time modeling based methods.

Index Terms: i-vector, multilingual, music genre classification, statistical feature

1. Introduction

Music is one of the most popular type of audio data on the Internet, and the amount of the on-line music data is huge. Therefore, it is increasingly needed to manage the music data in an effective way. One way is to automatically divide the music data into different classes, which is called the "music genre classification" [1].

The music genre classification is solved in the pattern recognition framework, and the main concentrations of conventional methods are in two aspects: feature extraction, which aims at finding more discriminative features, and classifier design, which focuses on designing better models to capture the discrimination of features belonging to different classes, and to generalize to unknown observations.

Feature extraction is a key factor to the performance of the classification system, and various techniques for feature extraction have been developed. Some works propose to combine different features to obtain a more comprehensive presentation of the audio than each individual feature. For instance, in [2], the mel-frequency cepstral coefficient(MFCC), rhythmic content features are jointly utilized to improve the classification performance. Some other studies transform feature into new spaces such that the transformed features are more discriminative in the new spaces. For example, in [3], the locality preserving non-negative tensor factorization and sparse representations are exploited. In addition, [4] suggests to use the long-term modulation spectral analysis of spectral features as well as the MFCC features.

The above mentioned methods generally perform on audio features extracted separately in each frame. However, as music data is sequential, these methods ignore the long-time time series information that is also important to music genre classification. Although [4] use a very long time widow about 340ms, it still cannot represent the music signal from the track level, and the feature in [4] have a very high dimension which is not conducive for the following modeling.

As we know that there are many important elements for music signal, such as loudness, rhythm, musical instrument, and so on. It is difficult to better capture so much elements in a music clip, and we use i-vector model in this paper to automatic learn these important elements for music signal. The GMM(Gaussian mixture model) in i-vector model can be seen as the combination of many Gaussian models, and each Gaussian model represents one element of music. The i-vector feature has been widely used in speaker recognition [5–8]. In this paper, the frame level features are used to train a i-vector extractor to get the segment level representation, and the GMM based universal Background model (GMM-UBM) is utilized to train the i-vector extractor. In addition, a multilingual i-vector model which combine multilingual model and i-vector model is further proposed to learn more about the elements using the unlabeled music data on the Internet (out-of-domain data). It has been recognized in automatic speech recognition (ASR) that the multilingual model built on the out-of-domain resource [9–11], can improve the performance of the target language. Experimental results show that the proposed model can improve the performance.

Although the i-vector has been widely used for speaker recognition tasks, there still lack effective methods to adopt it to music genre classification. To the best of our knowledge, this paper first uses i-vector for music genre classification. Further more, the multilingual i-vector can make use of large unlabeled music data on the Internet to improve the performance, and this has a great significance.

2. Data and Baseline Features

In this paper we choose the ISMIR database which has been widely used for music genre classification [4, 12–14]. The details of the database are described as Table 1, and the training set and testing set have been defined by the database. Six music genres are contained in the database which are classical, electronic, jazz/blue, metal/punk, rock/pop, and world respectively. The time lengths of different tracks are not fixed, and the total duration of all tacks is about 100 hours. Before feature extraction, each audio file has been converted into a 22050Hz, 16 bit, and single channel WAV file.

The Mel-Frequency Cepstral Coefficient (MFCC) is ob-

tained by averaging spectrogram values over mel-frequency. It is widely used in audio related tasks. We use MFCC as the low-level feature. We calculate the MFCC with 13 coefficients (including the energy coefficient) using a window of 25ms with 10ms overlap. In order to express the dynamic information, the first and second order derivatives are also computed. Finally, the feature is represented as 39-dimension MFCC.

Table 1. Database Description

genre	tracks(train/test)	time duration(hours)
Classical	320/320	17.87/16.71
Electronic	115/114	10.48/9.97
Jazz/Blue	26/26	1.66/1.80
Metal/Punk	45/45	3.14/2.95
Rock/Pop	101/102	6.34/6.79
World	122/122	11.92/10.86
total	729/729	51.41/49.08

But for music long time representation is beneficial, and MFCC will lose information when using long time window. So we need to explore the high level feature to better represent the music data. In this paper, the MFCC feature is used as the low-level feature to train high level feature which is called i-vector feature. In addition, the MFCC feature is also used as the baseline feature.

For comparing with the long-time representation (i-vector feature), the scattering feature is also explored. The scattering feature is proved success for music genre classification [15] [11]. It is an extension of MFCC, and it can recover the lost information by averaging spectrogram when using long time window. It is computed by scattering the signal information along multiple paths, with a cascade of wavelet modulus operators implemented in a deep convolution network (CNN).

In our work, we calculate first-order and second-order time scattering coefficients using a window of 370 ms with half overlap. The parameter for scattering transform are just the same as our previous work in [11]. In this paper, the scattering feature is also used as the baseline feature and the low-level feature.

3. I-Vector based Statistical Modeling

3.1. Framework

In this paper, the i-vector based statistical modeling method is proposed to represent music data and classify music genres. It contains three stages:In the first stage, we calculate low-level frame feature(scattering feature or MFCC feature) for music data as section 2.In the second stage, the low-level feature is used to get the high-level segment feature, we propose two high-level feature: the i-vector feature and multilingual i-vector feature, which will be described in detail in following two subsections.In the last stage, the high-level segment feature is used as the input of classification system to train the classification system and test the performance.

3.2. The I-vector Feature

The extraction of i-vector feature is shown in Fig .1. By using GMM-UBM, the original low-level feature is mapped to a high dimensional space to give a better representation of the audio track. Then, with total variability model(TVM) and Linear discriminant analysis(LDA), we reduce the dimension of the feature and finally obtain the i-vector feature.

The GMM-UBM is a kind of method to find the high dimensional space. GMM uses the probability estimation method

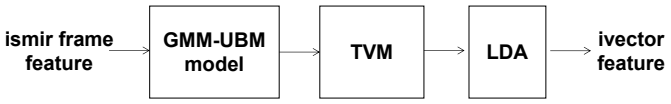

Figure 1: The extract of i-vector feature .

to describe the distribution of audio features. For a feature vector x, the probability density function is:

$$p(x|\lambda) = \sum_{i=1}^{C} \omega_i p_i(x) \tag{1}$$

$$\lambda = \{\omega_i, \mu_i, \Sigma_i\}, i = 1, 2, ..., C \tag{2}$$

where λ is the parameter set of GMM, C is the number of Gaussian mixture, D is the dimension of x, ω_i is the weight of each Gaussian mixture. We have $\sum_{i=1}^{C} \omega_i = 1$. μ_i is the mean value, and Σ_i is the covariance matrix. Each Gaussian component $p_i(x)$ can be represent as:

$$p_i(x) = \frac{1}{(2\pi)^{D/2}\Sigma_i^{1/2}} e^{-1/2(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)} \tag{3}$$

The GMM is trained by maximum likelihood estimation, and the local optimal solution is obtained by the expectation maximization (EM) algorithm. The posterior probability of x_k is:

$$\beta_i(x_k) = \frac{\omega_i p_i(x_k)}{p(x_k|\lambda)} \tag{4}$$

During the training of GMM, the zero order, first order and second order Baum-welch statistics (BWS) of the training feature $\{x_k|k = 1, 2, ..., N\}$ are computed by:

$$E_i^0 = \sum_{k=1}^{N} \beta_i(x_k) \tag{5}$$

$$E_i^1 = \sum_{k=1}^{N} \beta_i(x_k) x_k \tag{6}$$

$$E_i^2 = \sum_{k=1}^{N} \beta_i(x_k) x_k x_k^T \tag{7}$$

Then the GMM's parameter set λ is updated by following equations:

$$\omega_i = \frac{E_i^0}{N} \tag{8}$$

$$\mu_i = \frac{E_i^1}{E_i^0} \tag{9}$$

$$\omega_i = \frac{E_i^2}{E_i^0} - \mu_i \mu_i^T \tag{10}$$

Next, all BWS are concatenated to form a GMM-UBM super vector $m = [m_1^T, .., m_i^T, .., m_N^T]^T$ Next, all BWS are concatenated to form a GMM-UBM super vector:

$$m = [m_1^T, .., m_i^T, .., m_N^T]^T \tag{11}$$

where, $m_i = \frac{E_i^1}{E_i^0}$, and $1 \leq i \leq N$.

The dimension of GMM-UBM super vector is very high, and it makes the model tend to fall into the local optimal solution and increases the difficulty to train a stable model. Therefore, the dimension of the super vector is needed to be reduced.

We first use the TVM [16] to reduce the dimension. By applying factor analysis on the super-vector space, a vector m can be linearly projected into a low dimensional space as: $m = m_0 + Tw$, where m_0 is the mean value of m, T is a low-rank rectangular matrix, and w is the i-vector feature. Further details about the training of TVM can be seen in [16, 17].

LDA [16, 18, 19] is also widely used for dimension reduction. As the training of TVM is unsupervised, it doesn't contain the difference information of each music type. Therefore after the training of TVM, we use LDA to reduce the dimension in the supervised way. More information on LDA can refer to [16].

3.3. The Multilingual I-vector Feature

The extraction of multilingual i-vector feature is almost the same as that of i-vector feature with the difference only in the training of GMM-UBM. In the multilingual i-vector based model, we train the GMM-UBM using large out-of-domain music data an ISMIR database. It is well known that more data helps to train a more robust system. However, in our ISMIR database, the amount of music data is not enough especially for some genres. In the multilingual i-vector, we find a large amount of unlabeled music data from the Internet, and train the GMM-UBM in an unsupervised manner.

Fig. 2 shows the procedure of using unlabeled out-of-domain music data to train the GMM-UBM model. The out-of-domain music data is randomly download from Baidu Music website [20], which contains 4550 music tracks with total duration about 304 hours. The language of the those music tracks includes Chinese, English, Japanese, French, etc. The training of GMM-UBM and TVM is just the same as the previous subsection. After that we get the multilingual i-vector feature, and then the LDA is used to reduce the dimension.

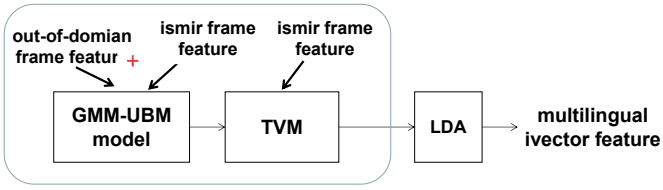

Figure 2: The extract of multilingual i-vector feature .

4. Experiment and Analysis

4.1. The Baseline Systems

DNN based systems are built as the baselines. The frame-level feature (MFCC feature or scattering feature) is utilized as DNN input. The DNN has one input layer, one output layer, and several hidden layers which will be elaborated in the following paragraphs. Within the hidden layers, the input of each node is computed as the linear combination of the outputs from the previous layer. Each hidden node transforms its input with a sigmoid activation function to achieve non-linearity. Then, a soft-max output function is used at the output layer to compute the posterior probability. At last, the mean square error cost function is used to train the DNN.

The toolkit for DNN training is the Karel's DNN in Kaldi Toolkit [21]. The learning rate is $8 * 10^{-6}$, and the training epochs for training the DNN is 100. In the testing stage, the label of each frame is first recognized and then majority voting is used to get the track labels.

We build two baseline systems using different features. The first baseline system is based on the MFCC feature, and the second baseline system is based on the scattering feature. The structure of DNN and the result are list in Table 2, which also lists the performance of other existing approaches. In Table 2, "39" and "525" are the number of nodes in the input layer and "6" is the number of nodes in the output layer.

Table 2. The results of baseline systems

baseline models(layers)	describe	accuracy
DNN1(39-128-6)	MFCC+DNN	43.21%
DNN1(39-512-6)	MFCC+DNN	42.80%
DNN1(39-512-512-6)	MFCC+DNN	42.94%
DNN1(39-128-512-128-6)	MFCC+DNN	43.35%
DNN2(525-1024-6)	Scatt+DNN	85.32%
DNN2(525-1024-1024-6)	Scatt+DNN	84.35%
hol2008 [22]	existing work	83.5%
lee2009 [4]	existing work	86.8%
sig2014 [13]	existing work	73.4%

4.2. The Experiment of I-Vector based Statistical Modeling Method

The procedure of computing the i-vector representation has been described in the previous sections. The training tool we used to get the i-vector is Kaldi Toolkit [21]. As the i-vector representation is the track-level feature, the amount of feature is not enough to train a DNN. So with the i-vector representation, we use linear SVM as the classifier. The result is summarized in Table 3. In Table 3, "ubm512" means that the number of Gaussian mixture in GMM-UBM is 512, "TVM200" means that the TVM model reduce the feature dimension into 200, "LDA50" means that we use LDA to reduce the feature dimension into 50, and "multi" means using multilingual based i-vector representation.

Table 3. The results of different models

model name	feature	accuracy
ubm512+TVM200	Scatt	28.15%
ubm512+TVM400	Scatt	43.89%
ubm512+TVM200	MFCC	86.97%
ubm512+TVM400	MFCC	87.38%
ubm512+TVM400+LDA50	MFCC	88.48%
ubm512+TVM400+LDA200	MFCC	88.48%
ubm512+TVM400+LDA300	MFCC	88.34%
multi+ubm512+TVM400+LDA50	MFCC	89.99%
multi+ubm512+TVM400+LDA200	MFCC	90.12%

From Table 3, we can see that the classification system using scattering feature has a poor performance. That because scattering feature has a high dimension (525 dimension) which is not appropriate for GMM training. It is obvious that LDA can improve the classification performance, and the multilingual i-vector representation perform further better than others.

4.3. Experiment Analysis

In this subsection, we first analysis the effect of parameters in GMM-UBM and TVM. Fig. 3(a) is the accuracy of different number of Gaussian mixtures in GMM-UBM using different models. From the Fig. 3(a), we can see that the number of

Gaussian mixture from 512 to 2048 has little effect on the experimental results. Fig. 3(b) is the accuracy of different reduced dimension using TVM model. Fig. 3 (c) is the accuracy of different reduced dimension using LDA. From the Fig. 3(b) and Fig. 3(c), we can see that "TVM+LDA" performs better than only using "TVM", which shows that LDA makes the feature more appropriate for classification. The results also show that the multilingual i-vector based model is better than i-vector based model, and indicate that using the out-of-domain database for unsupervised training can improve the classification performance.

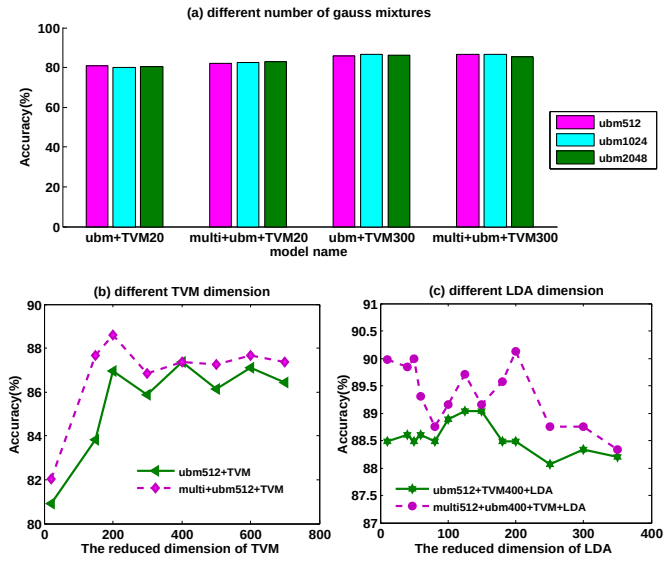

Figure 3: In the Fig, "ubm" means GMM-UBM model is trained only use ISMIR database. "multi+ubm" means GMM-UBM model is trained on ISMIR database and large out-of-domain database from the Internet. "ubm512" means the gauss mixture of GMM-UBM is 512. "TVM300" means reduce the feature dimension into 300 using TVM. The detail describe of different models are in section 3.

5. Conclusion

This paper investigates i-vector based statistical modeling for music genre classification. The i-vector transforms the frame feature to statistical segment level representation. Then we combine the i-vector model with the multilingual model to get multilingual i-vector feature. This method enables us to use the unlabeled data on the Internet, and experimental results show the superiority of the proposed models. As we have a large amount of unlabeled data on the Internet, and if we can make use of these data to improve our system, it will be a significant improvement.

6. Acknowledgements

This research was supported by following two parts: The China National Nature Science Foundation (No. 61573357, No. 61503382No. 61403370, No. 61273267 and No. 91120303, No. 61305027); Technical development project of state grid corporation of China entitled "machine learning based Research and application of key technology for multi-media recognition and stream processing".

7. References

[1] X. Yang, Q. Chen, S. Zhou, and X. Wang, "Deep belief networks for automatic music genre classification," *ntM*, vol. 92, no. 11, pp. 2433–2436, 2011.

[2] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[3] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Music genre classification using locality preserving non-negative tensor factorization and sparse representations," in *ISMIR*, 2009, pp. 249–254.

[4] C. H. Lee, J. L. Shih, K. M. Yu, and H. S. Lin, "Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features," *Multimedia, IEEE Transactions on*, vol. 11, no. 4, pp. 670–682, 2009.

[5] W. Zhu and J. Pelecanos, "Online speaker diarization using adapted i-vector transforms," in *International Conference on Acoustics, Speech, and Signal Processing*, 2016.

[6] L. F. Gallardo, M. Wagner, and S. Moller, "I-vector speaker verification based on phonetic information under transmission channel effects," in *Interspeech*, 2014.

[7] P. Matejka, O. Glembek, F. Castaldo, and M. J. Alam, "Full-covariance ubm and heavy-tailed plda in i-vector speaker verification," in *International Conference on Acoustics*, 2011, pp. 4828–4831.

[8] O. Glembek, L. Burget, P. Matejka, and M. Karafiat, "Simplification and optimization of i-vector extraction," vol. 125, no. 3, pp. 4516–4519, 2011.

[9] F. Grézl and M. Karafiát, "Combination of multilingual and semi-supervised training for under-resourced languages," in *Fifteenth Annual Conference of the International Speech Communication Association*, vol. 2014, no. 9. International Speech Communication Association, 2014, pp. 820–824.

[10] N. T. Vu, Y. Wang, M. Klose, Z. Mihaylova, and T. Schultz, "Improving asr performance on non-native speech using multilingual and crosslingual information," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[11] J. Dai, W. Liu, C. Ni, L. Dong, and H. Yang, "Multilingual deep neural network for music genre classification," in *Interspeech*, 2015.

[12] F. D. Leon and K. Martinez, "towards efficient music genre classification using fastmap," *Proc Dafx*, 2012.

[13] S. Sigtia and S. Dixon, "Improved music feature learning with deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 6959–6963.

[14] "Ismir 2004 music genre dataset," http://ismir2004.ismir.net/genre_contest/index.html.

[15] J. Andén and S. Mallat, "Deep scattering spectrum," *CoRR*, vol. abs/1304.6763, 2013.

[16] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *Trans. Audio, Speech and Lang. Proc.*, vol. 19, no. 4, pp. 788–798, May 2011.

[17] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice modeling with sparse training data," *IEEE Transactions on Speech & Audio Processing*, vol. 13, no. 3, pp. 345–354, 2005.

[18] W. S. Zheng, J. H. Lai, and S. Z. Li, "1d-lda vs. 2d-lda: When is vector-based linear discriminant analysis better than matrix-based?" *Pattern Recognition*, vol. 41, no. 7, pp. 2156–2172, 2008.

[19] S. F. Hafez, M. M. Selim, and H. H. Zayed, "2d face recognition system based on selected gabor filters and linear discriminant analysis lda," *Computer Science*, vol. 12, pp. 33–41, 2015.

[20] "Baidu music," http://music.baidu.com/.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, and P. Schwarz, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 2011.

[22] A. Holzapfel and Y. Stylianou, "Musical genre classification using nonnegative matrix factorization-based features," *IEEE Transactions on Audio Speech and Language Processing*, vol. 16, no. 2, pp. 424–434, 2008.