Vocal Language Classification

Xinquan Zhou
902998187

April 27, 2014

Extra credits

There are totally 24 (3+8+13) questions in CIOS!

Task

Automatically detecting the language in the music songs is an interesting topic in the Music Information Retrieval (MIR) and is not widely studied for the moment. Although lots of research has focused on the spoken language identification task and the accuracy of this task can be kind of satisfying in many cases. Vocal sound language identification in music is different and more difficult in some sense than spoken language. It is because most of the speech classification methods are based on phoneme recognition which is extremely hard to be detected in music since the pitch varies much from the normal speech. Besides pitch, the duration of the phoneme also can be very different between music and speech. My goal of this project is that building a proper classifier to automatically classify which language of the input music song, the input song can be both live humming and recording.

Motivation

Vocal language classification is an interesting topic in MIR and has many user cases. With the development of international cooperation and communication, language identification tends to be increasingly significant for music technology industry. For example, it can serve for query by humming and music recognition. Also it can involve music robotic in terms of interaction. And I believe there are lots of potential usages to be digged out.

Related Work

In the last tens of years, there has been much research working on this task. First of all, Gaussian Mixture Model (GMM)[1] for language identification (LID) served as the simplest way of this study in which each language is modeled as a GMM to do the classification. Furthermore, phone recognition followed by language modeling (PRLM)[2], which is a single-language phone recognizer followed by an n-gram analyzer, and instead of single language model in PRLM, parallel PRLM exploiting Multilanguage model became effective methods for this task. More recently, Campell and etc[3] exploit the ability of SVMs in which they use a kernel that compares sequences of feature vectors and produces a measure of similarity for language recognition. SantoshKumar and Ramasubramanian[4] establish the equivalence of an ergodic-Hidden Markov Model (EHMM) to a parallel sub-word recognition framework for spoken language identification which gives me much inspiration on my research. Compared with spoken language identification, vocal sound language classification in music has not widely studied yet. Wei-Ho Tsai and Hsin-Min Wang[5] present a first attempt to automatically identify the language sung in a music recording. And Jochen Schwenninger and etc[6] make efforts of transferring well-established techniques from spoken language identification to the area of language identification in music.

Dataset and Features

There seems not be the public standard dataset for this task, so that I use my personal collection of popular songs of English and Chinese as part of the dataset for the research. There are 90 different English singers and 50 different Chinese singers in my collection. And I select 2 to 3 songs on average of each singers building the dataset. The number of English songs is 180, while the number of Chinese is 124, each song last 4 minutes on average. And I also collect 110 English vocal songs from QUASI Database and MTG-QBH dataset, and 150 Chinese vocal songs from MIR-1K Dataset. So totally I have 290 English songs and 274 Chinese songs. According to the feature study in spoken language identification by Rong Tong[7] and Weiqiang Zhang[8], mel-frequency cepstral coefficients (MFCC) and Shifted Delta Coefficients (SDC) are very suitable for this particular task. So I extract 13 dimensions MFCC and 49 dimensions SDC building 62 dimensions feature vector.

Methodology

Preprocessing

In the music, background music has much bad effects on the accuracy of language classification, therefore what I want is just the pure vocal part in the songs. In order to get the singing voice, I use source separation method of Po-Sen Huang and etc[9] to process the pop music in my collection. Then, noticing that there are much silent part in some samples, I decide to do the voice detection based on [10] for each sample and cut off the silent part. And I calculate 13 MFCC and 49 SDC for each frame of each sample. And frame length is 50ms and hopsize is 25ms. So each sample is represented as a m × n matrix, where m is the dimension of features which is 62, n is the number of frames in the sample. Because the value range of SDC and MFCC differs a lot, I do the normalization using Z score for all the samples. Lots of classifiers can not use this kind of time series directly, so for these classifiers, I combine 40 adjacent frames using a slide window and calculate the mean to create new sub-samples and input these sub-samples into classifiers.

Classifiers in Weka[11]
SVM(SMO): Using c = 1.0, RBFkernel
NaiveBayes: With useKernelEstimator to be false
Random Forest: With the 50 trees
AdaBoostM1: With 50 iterations
1-NN

EHMM

`EHMM System`:
One of the earliest work in LID by House and Neuberg[12] was based on the now popular hidden Markov model (HMM). Following this, there have been a few other attempts to use HMMs for LID. According to [6] who established the equivalence of the parallel sub word recognition framework to an EHMM. EHMM is the fully connected HMM as shown as figure 1. The states of EHMM correspond to sub-work units and the state-transition probabilities represent the bigram statistics of language model. And the state observation densities are represented by GMM. I use basically the same model as above. I train one EHMM $L_1$to L_N for every target language. And given an input audio, I calculate the likelihood of each EHMM for the input using Viterbi decoding, and output the language which has the highest likelihood V_i, i.e. $\tilde{i} = argmax_{i=1,...,N}(V_i)$. The system float chart is shown as figure 2.

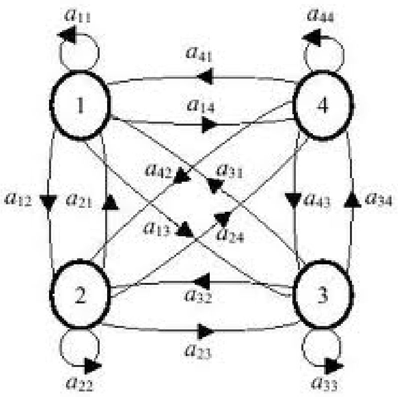

Figure 1: Ergodic HMM

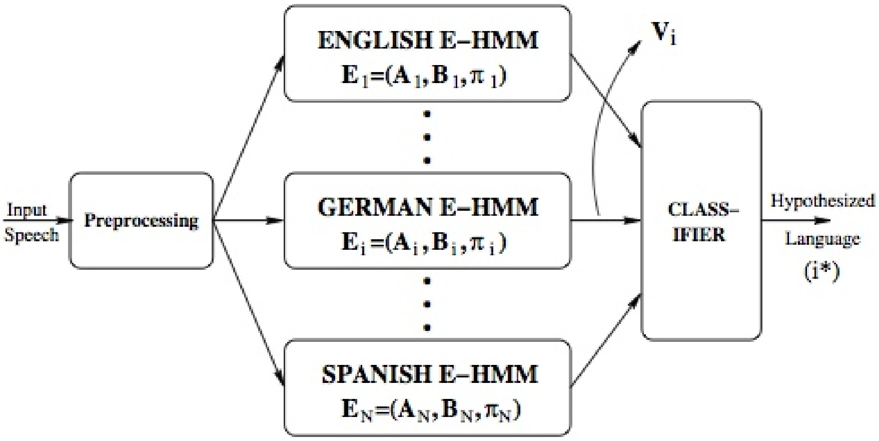

Figure 2: System Float Chart

`Parameter Specification`:
An M-state EHMM of language is specified as $L_i = (A_i, B_i, \pi_i)$,where A_i is the state transition probability, B_i is state observation probability and π_i is the initial state probability. We have to initialize the state transition probability, initial state probability, and state observation probability before using the EHMM. Usually A_i and π_i can be randomly selected or uniformly set. I uniformly set these values. As for B_i I use standard K-means to initialize it: Firstly

I cluster each language feature vectors into several groups that are corresponding the initial states. And then I calculate the mean and covariance matrix of each group. Therefore I assign these mean and covariance matrix to the initial parameters of B_i. To avoid the covriance of GMM to be ill-conditioned, I assume the features are dependent to each other so that using diagonal coariance matrix.

Evaluation
I use 5-fold cross validation to access the classifiers. For better evaluation, I compare the confusion matrix and calculate accuracy of the results.

1 Experiment and Results

I present here experimental results using SVM (SMO), Naive Bayes, Random Forest, AdaBoostM1 and 1-NN. As talked above, the total number of samples is 169452. This is the experiment 1. And then I use the features extracted from original songs without singing voice separation forming 158031 samples to do the experiment again, and this is the experiment 2. The results are shown in table 1.

Table 1: The accuracy of different classifiers

Classifier	SVM(SMO)	Naive Bayes	AdaBoostM1	Random Forest	1-NN	EHMM
Accuracy for Ex.1	0.6184	0.5801	0.6278	0.6347	0.6544	0.9274
Accuracy for Ex.2	0.6649	0.6098	0.6507	0.7357	0.6571	0.8052

2 Analysis and Conclusion

The result of EHMM is much better than the others. It can be explained that on the one hand, audio signal is a time series stream, so EHMM can capture the properties much better and accurate. On the other hand, maybe maybe only taking the mean is not good way to represent the samples. It can lose lots of information. And the window size(bonding adjacent frames together to extract features) is too small. Looking into the features, I find almost every pair of features are kind of uncorrelated or related in the strange way. For example the figure 3, 4, and 5 illustrates the typical relation between some pairs of features, the colors represent classes. Almost all pairs of features have similar relation as shown in these figures. It is really hard to imagine how they distribute in the 62-dimension space. Thus I think it will be extremely hard to find a hyperplane to separate them well. So in a another way of thinking: maybe the values of feature may not matter a lot but the dynamic changing does. And HMM can represent this changing with the transition probability. Furthermore, the states in HMM have kind of physical meanings of the language: the sub-work unit. Therefore using HMM is more intuitive for this task as well. And in the real language, the combination of the words can be really large, so using EHMM with fully connected states can precisely indicate this characteristic of language. Additionally, we find that after using source separation songs as training performs worse than using the original songs except for EHMM classifier. It may be because the source separation algorithm is not good enough to extract pure vocal sound, therefore introducing many similar artifacts so that making the class even more vague and hard to distinguish.

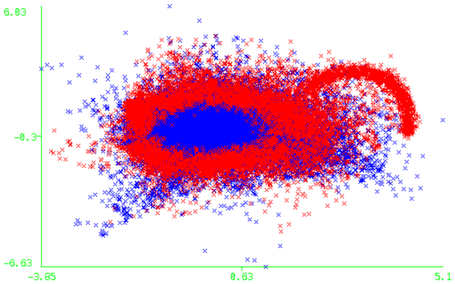

Figure 3: Plot for feature 2 and 3

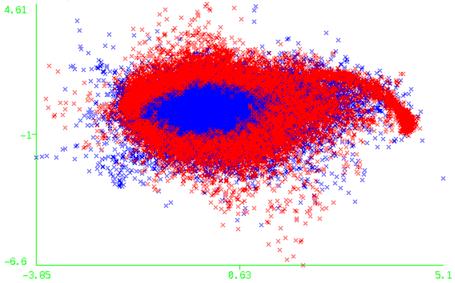

Figure 4: Plot for feature 2 and 4

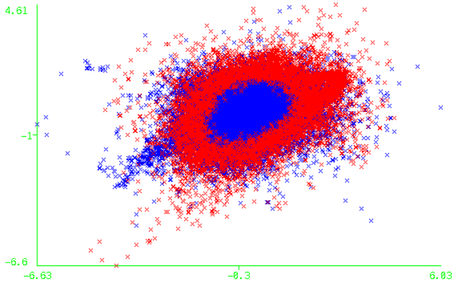

Figure 5: Plot for feature 3 and 4

References

[1] M. A. Zissman, "Automatic language identification using gaussian mixture and hidden markov models," in *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, vol. 2, pp. 399–402, IEEE, 1993.

[2] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, p. 31, 1996.

[3] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Computer Speech & Language*, vol. 20, no. 2, pp. 210–229, 2006.

[4] S. SantoshKumar and V. Ramasubramanian, "Automatic language identification using ergodic-hmm," 2005.

[5] W.-H. Tsai and H.-M. Wang, "Towards automatic identification of singing language in popular music recordings.," in *ISMIR*, Citeseer, 2004.

[6] J. Schwenninger, R. Brueckner, D. Willett, and M. E. Hennecke, "Language identification in vocal music.," in *ISMIR*, pp. 377–379, 2006.

[7] R. Tong, B. Ma, D. Zhu, H. Li, and E. S. Chng, "Integrating acoustic, prosodic and phonotactic features for spoken language identification," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1, pp. I–I, IEEE, 2006.

[8] W. Zhang, J. Liu, and L. He, "Auditory features with vocal track length normalization for language identification," in *Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on*, pp. 66–70, IEEE, 2008.

[9] P.-S. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson, "Singing-voice separation from monaural recordings using robust principal component analysis," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 57–60, IEEE, 2012.

[10] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *Signal Processing Letters, IEEE*, vol. 6, no. 1, pp. 1–3, 1999.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[12] A. S. House and E. P. Neuburg, "Toward automatic identification of the language of an utterance. i. preliminary methodological considerations,"

The Journal of the Acoustical Society of America, vol. 62, no. 3, pp. 708–713, 1977.