HOMOGENEOUS SEGMENTATION AND CLASSIFIER ENSEMBLE FOR AUDIO TAG ANNOTATION AND RETRIEVAL

Hung-Yi Lo, Ju-Chiang Wang, Hsin-Min Wang

Institute of Information Science, Academia Sinica, Taipei, Taiwan
Email: {hungyi, asriver, whm}@iis.sinica.edu.tw

ABSTRACT

Audio tags describe different types of musical information such as genre, mood, and instrument. This paper aims to automatically annotate audio clips with tags and retrieve relevant clips from a music database by tags. Given an audio clip, we divide it into several homogeneous segments by using an audio novelty curve, and then extract audio features from each segment with respect to various musical information, such as dynamics, rhythm, timbre, pitch, and tonality. The features in frame-based feature vector sequence format are further represented by their mean and standard deviation such that they can be combined with other segment-based features to form a fixed-dimensional feature vector for a segment. We train an ensemble classifier, which consists of SVM and AdaBoost classifiers, for each tag. For the audio annotation task, the individual classifier outputs are transformed into calibrated probability scores such that probability ensemble can be employed. For the audio retrieval task, we propose using ranking ensemble. We participated in the MIREX 2009 audio tag classification task and our system was ranked first in terms of F-measure and the area under the ROC curve given a tag.

***Keywords*—**audio segmentation, audio tag annotation, audio tag retrieval, ensemble method

1. INTRODUCTION

With the explosive growth of digital music available on the Web, organizing and retrieving desirable music from online music databases on the Web becomes an increasingly important and challenging task. Traditionally, some music information retrieval (MIR) research was focused on musical information classification with respect to genre, mood, instrumentation, quality, etc. Recently, social tags have started to play a key role in the development of "Web 2.0" technologies and have become a major source of information for recommendation. Music tags are free text labels associated with artists, genre, emotion, mood, instruments, etc [1]. Consequently, music tag classification seems to be a more complete and practical way for musical information classification. Given a music clip, we hope the tagging algorithm can automatically predict tags for the music clip based on the models trained from music clips with associated tags collected beforehand.

Recently, automatic audio tag annotation has been a raising and active research topic [2-5]. It has been one of the evaluation tasks in Music Information Retrieval Evaluation eXchange (MIREX) since 2008 [1]. The best audio tag annotation and retrieval system [3] in MIREX 2008 models the feature distribution for each tag with a Gaussian mixture model and estimates the model parameters with the weighted mixture hierarchies expectation maximization algorithm. A more recent work [5] uses the Codeword Bernoulli Average (CBA) model with vector quantized feature representation. In contrast to using probability models, Eck et al. [2] use AdaBoost to automatically generate audio tags for music recommendation.

This paper evaluates our method for the audio tagging problem in two aspects: audio tag annotation and audio tag retrieval. The audio annotation task is considered as a binary classification problem of each tag, since a fixed number of tags are given. In other words, we train a binary classifier for each tag. Each tag classifier verifies whether the input audio clip should have the specific tag or not by outputting a score. The performance can be evaluated in terms of the percentage of tags that are verified correctly or the area under the ROC curve (AUC-ROC) given a clip (i.e., given a clip, correct tags should receive higher scores). For the audio retrieval task, given a specific tag as the query, we want to retrieve the audio clips that are corresponding to the tag. This can be done by using the tag classifier to determine the score that each audio clip is relevant to the tag. The clips will be ranked with the relevance scores; therefore, the clips with higher scores will be returned to the user. The performance can be evaluated in terms of the tag F-measure or the AUC-ROC given a tag (i.e., given a tag, correct clips should receive higher scores).

This work was supported in part by Taiwan e-Learning and Digital Archives Program (TELDAP) sponsored by the National Science Council of Taiwan under Grant: NSC99-2631-H-001-020.

[1] http://www.music-ir.org/mirex/2008/

The major contributions of this work are as follows. First, we propose dividing the audio clip into several homogeneous segments by using an audio novelty curve [6]. In [7], an audio clip was simply partitioned into several *fixed length* segments for music genre classification. Second, we exploit an ensemble classifier, which consists of Support Vector Machine (SVM) and AdaBoost classifiers, for tag classification. We participated in the MIREX 2009 audio tag classification task and our system was ranked first in terms of tag AUC-ROC and F-measure, compared to the other submissions, including the CBA method [5]. Third, we propose transforming the output scores of the component classifiers into calibrated probability scores such that they can be easily combined by the classifier ensemble. This step can improve the performance in terms of clip AUC-ROC and tag accuracy.

The remainder of this paper is organized as follows. In Section 2, we give an overview of our method. Then, we describe feature extraction and audio segmentation in Section 3, and present our classification method in Section 4. The experiments and results are detailed in Section 5. Finally, the conclusions are drawn in Section 6.

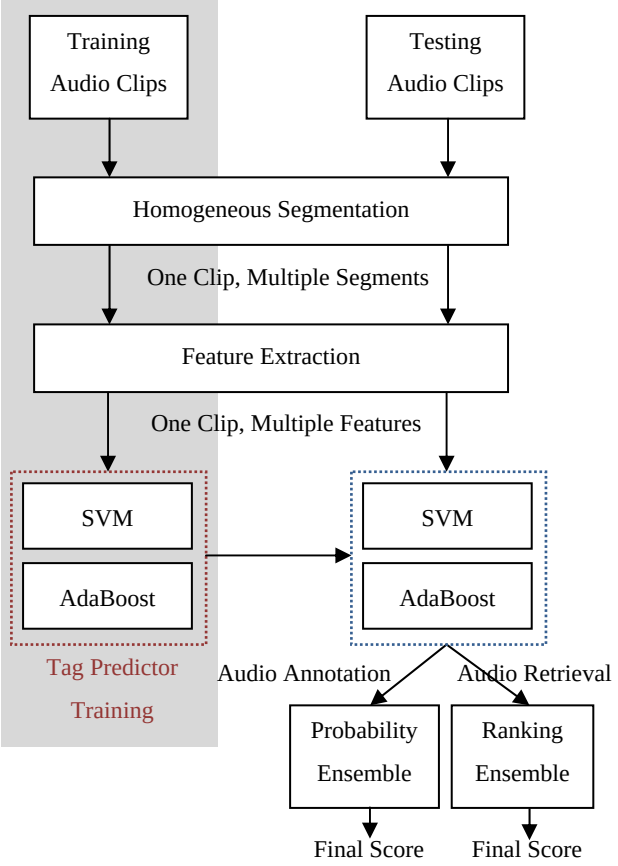

Fig. 1. The work flow of the proposed audio tag annotation and retrieval system.

2. SYSTEM OVERVIEW

Figure 1 shows the work flow of our system. We first split an audio clip into homogeneous segments, and then extract audio features with respect to various musical information, including dynamics, rhythm, timbre, pitch, and tonality, from each segment. The features in frame-based feature vector sequence format are further represented by their mean and standard deviation such that they can be combined with other segment-based features to form a fixed-dimensional feature vector for a segment. The prediction score for an audio clip given by a classifier is the average of the scores for its constituent segments. In the training phase, we train an ensemble classifier, which consists of SVM and AdaBoost classifiers, for each tag. In the testing phase, the scores of the component classifiers are merged by using probability ensemble for annotating an audio clip or ranking ensemble for ranking all the audio clips given a tag.

3. FEATURE EXTRACTION

For applying machine learning techniques to audio tag classification, we need to extract characteristic features of various types from the waveform of an audio clip by using some signal processing methods. Since feature selection is embedded in the training process of our classification method, we extract as many kinds of features as possible. However, for some frame-based features, such as Mel-frequency cepstral coefficients (MFCCs), we need to convert the variable-length feature vector sequence into a fixed-dimensional feature vector such that they can be used jointly with other features, like key and tempo. In this paper, the frame-based features are represented by their mean and standard deviation.

It is very likely that only a portion of the audio clip is associated with a specific tag. For instance, an audio clip may have the tag “female vocal” even though a female vocal only appears in the front part of the clip. Therefore, it might be inadequate to use the mean of MFCC vectors to represent the timbre of the whole clip. To solve this problem, we divide the clip into homogeneous segments and treat each segment as a unit in tag classification. Then, the final decision for the clip is based on the fusion of the results of its constituent segments.

3.1. Feature Extraction

For feature extraction, we use MIRToolBox 1.1[2], which is a free and powerful MATLAB tool for MIR tasks. The detailed descriptions of the audio features supported by the tool can be comprehended in its user’s manual [8]. The

[2] http://users.jyu.fi/~lartillo/mirtoolbox/

features used in this paper are categorized into five fields: dynamics, rhythm, timbre, pitch, and tonality. They include:

- in the dynamics field: rms;
- in the rhythm field: (1) the peak and centroid of the fluctuation summary, (2) tempo, and (3) attack slop and attack time of the onset;
- in the timbre field: (1) zero-crossing rate, (2) spectral centroid, spread, skewness and kurtosis, (3) brightness, (4) rolloff with 95% threshold, (5) rolloff with 85% threshold, (6) spectral entropy and flatness, (7) roughness, (8) irregularity, (9) inharmonicity, (10) MFCCs, delta-MFCCs, and delta-delta-MFCCs, (11) low energy rate, and (12) spectral flux;
- in the pitch field: (1) pitch, (2) chromagram and its centroid and highest peak; and
- in the tonality field: (1) key clarity, (2) key mode, and (3) harmonic change.

The audio clip is in format 16-bit, 44.1kHz, and stereo. We apply the default parameters, such as the length of window and hop size, in MIRToolBox for feature extraction. After feature extraction, each clip (or segment as will be discussed later) is represented by a 174-dimensional feature vector.

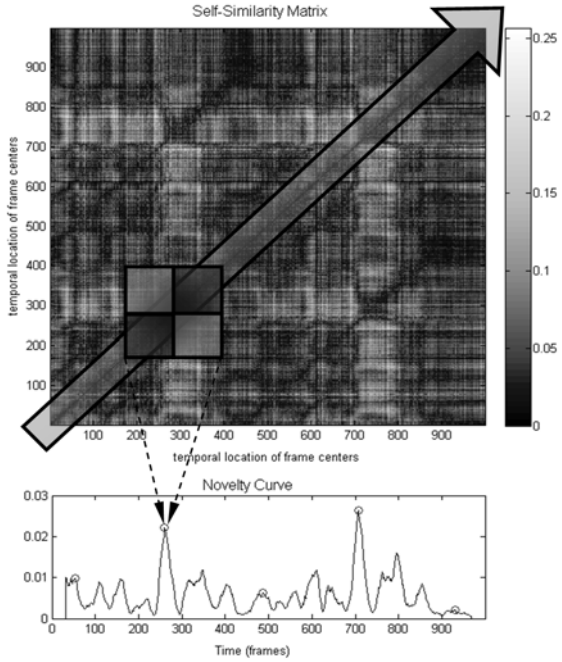

Fig. 2. An illustration of audio segmentation.

3.2. Audio Segmentation

Our audio segmentation is based on a measure of audio novelty proposed in [6]. We compute the cosine distance of MFCC vectors between any pairs of two frames in the audio clip, and build a self-similarity matrix, which can be visualized as a square image in the top panel of Figure 2. The gray scale value of a pixel in the image is proportional to the cosine distance. Then, we can obtain a time-aligned novelty curve, as shown in the bottom panel of Figure 2, by sliding a checkerboard kernel with a radial Gaussian taper along the diagonal of the similarity matrix. Finally, we apply a weighted dynamic thresholding scheme [9] to locate the local peaks of the novelty curve as segment boundaries. The local peaks are marked by circles in the bottom panel of Figure 2. To prevent feature extraction and classification failures caused by insufficient data, we limit the length of each segment to be at least 0.5 seconds. The resulting number of segments for each 10-second clip is from 2 to 5.

4. THE CLASSIFICATION METHOD

In this section, we discuss our classification method. Since each audio clip can have multiple tags, following the works in [2, 4, 7] that assume the tags are independent, we transform the tag prediction problem into many independent binary classification problems, each for an individual tag. For each tag, our final prediction combines the outputs of two classifiers: Support Vector Machine (SVM) and AdaBoost.

4.1. Support Vector Machine

SVM is one of the most promising learning algorithms for the classification problem and has been successfully applied to the music classification task [10]. SVM finds a separating surface with a large margin between training samples of two classes in a high dimensional feature space implicitly introduced by a computationally efficient kernel mapping, and the large margin implies good generalization ability according to the statistical learning theory. In this work, we exploited a linear SVM classifier $f(\mathbf{x})$ of the following form:

$$f(\mathbf{x}) = \sum_{j=1}^{m} w_j x_j + b \,, \tag{1}$$

where x_j is the j-th feature of the feature vector $\mathbf{x}$ of a test sample; w_j and b are parameters to be trained from a training set $\{(\mathbf{x}_i, y_i), i=1,\ldots,n\}$, where $\mathbf{x}_i$ is the feature vector of the i-th training sample and y_i is the class label. The advantage of using the linear SVM is its training efficiency, and some recent literatures show that it has comparable prediction performance to the non-linear SVM. A single cost parameter C is determined by using cross-validation and the selection of C will be discussed in Section 5.

4.2. AdaBoost

Boosting is a method of finding a highly accurate classifier by combining many base classifiers, each of which is only moderately accurate. AdaBoost has also been successfully used in applications such as music classification [7] and audio tag classification [2]. We use decision stumps as the base learner. The decision function of the boosting classifier takes the following form:

$$g(\mathbf{x}) = \sum_{t=1}^{T} \alpha_t h_t(\mathbf{x}), \quad (2)$$

where α_t is set as suggested in [11]. The model selection procedure can be done efficiently as we can iteratively increase the number of base learners and stop when the generalization ability on the validation set does not improve.

4.3. Ranking Ensemble

We noticed that the scales of the two classifiers' prediction scores are rather different. Given a batch of testing clips, we first rank the prediction scores of individual classifiers independently. Then, the final score for a clip is the average of the ranks from the two classifiers. In this way, the smaller the average rank, the more likely the audio clip has the specific tag. We have applied this method in our MIREX 2009 submission. It achieves very good performance in terms of tag F-measure and tag AUC-ROC as these two metrics are more related to the ranking performance. However, the performance in terms of clip AUC-ROC is poor. In fact, this method is not suitable for the audio annotation task because it is unpractical to annotate a clip by referring to other clips simultaneously. In order to annotate a single clip, we need to combine the scores from the two classifiers in a different way. Therefore, we propose probability ensemble instead of ranking ensemble for the audio annotation task.

4.4. Calibrated Probability Scores and Probability Ensemble

As each tag classifier is trained independently, the raw scores of different tag classifiers are not comparable. In the audio annotation task, we need to compare the scores of all tag classifiers and determine which tags should be associated with the given audio clip. Therefore, we transform the output score of SVM and AdaBoost into a probability score with a sigmoid function [12]:

$$\Pr(y = 1 \mid \mathbf{x}) \approx \frac{1}{1 + \exp(Af + B)}, \quad (3)$$

where f is the output score of a classifier, A and B are learned by solving a regularized maximum likelihood problem as suggested in [13]. As the classifier output has been calibrated into a probability score, a classifier ensemble is formed by averaging the probability scores of SVM and AdaBoost, and the probability scores of different tag classifiers become comparable.

5. MIREX 2009 RESULTS AND EXTENDED EXPERIMENTS

5.1. MIREX 2009 Results

The submissions to the MIREX 2009 audio tag classification task have been evaluated on two datasets: the MajorMiner set and the mood set [14]. The algorithms were evaluated with three-fold cross validation and artist filtering was used in the production of the test and training splits. The evaluation metrics include the tag F-measure and the area under the receiver operating characteristic curve given a tag (tag AUC-ROC). Both metrics are corresponding to the tag retrieval task that is aimed at retrieving audio by a given tag query. The metrics also include the AUC-ROC given a clip (clip AUC-ROC) and the tag accuracy. These two metrics are corresponding to the tag annotation task that is aimed at annotating a given audio clip with correct tags.

Table 1. Evaluation results of MIREX 2009 audio tag classification on the MajorMiner dataset.

	Tag F-measure	Tag Accuracy	Tag AUC-ROC	Clip AUC-ROC
NOS	0.289	0.900	0.782	0.751
SEG	**0.311**	0.903	**0.807**	0.774
A1	0.277	0.868	0.742	0.871
A2	0.290	0.859	0.761	0.861
B1	0.209	0.912	0.762	**0.882**
B2	0.241	0.905	0.791	**0.882**
B3	0.170	0.913	0.721	0.854
B4	0.263	0.890	0.749	0.854
C	0.012	0.891		
D1	0.290	0.850	0.784	0.872
D2	0.293	0.850	0.786	0.876
E	0.044	**0.914**	0.736	0.851

Table 2. Evaluation results of MIREX 2009 audio tag classification on the mood dataset.

	Tag F-measure	Tag Accuracy	Tag AUC-ROC	Clip AUC-ROC
NOS	0.204	0.882	0.667	0.678
SEG	**0.219**	0.887	**0.701**	0.704
A1	0.195	0.837	0.648	0.854
A2	0.193	0.829	0.632	0.859
B1	0.172	0.878	0.652	0.849
B2	0.180	0.882	0.681	0.848
B3	0.147	0.882	0.629	0.812
B4	0.183	0.862	0.646	0.812
C	0.084	0.863		
D1	0.211	0.823	0.649	0.860
D2	0.209	0.824	0.655	**0.861**
E	0.063	**0.909**	0.664	**0.861**

The results of evaluation on the two datasets are summarized in Tables 1 and 2, respectively. The best result of each specific evaluation metric is bold-typed. The names in the first column indicate the twelve submissions. Our submissions without and with pre-segmentation are denoted by NOS and SEG, respectively. It is clear that pre-segmentation is effective. Table 3 summarizes the ranking of our two submissions in terms of the four evaluation metrics on the two datasets. Our SEG submission achieves the best performance in terms of the metrics corresponding to the audio retrieval task (i.e., tag F-measure and tag AUC-ROC) but performs poorly in terms of the metric corresponding to the audio annotation task (i.e., clip AUC-ROC). The details about the evaluation datasets and the other submissions are available on the MIREX website[3].

Table 3. Performance rankings of our two submissions to MIREX 2009 audio tag classification on two datasets.

	Evaluation Metrics	Ranking	
		SEG	NOS
The MajorMiner Dataset	Tag AUC-ROC	1	5
	Tag F-measure	1	5
	Clip AUC-ROC	11	12
	Tag Accuracy	5	6
The Mood Dataset	Tag AUC-ROC	1	3
	Tag F-measure	1	4
	Clip AUC-ROC	11	12
	Tag Accuracy	2	3

5.2. Extended Experiments

This subsection presents the results of extended experiments on the downloaded MajorMiner dataset. We extensively evaluate the SVM classifier, the AdaBoost classifier, the ranking ensemble method, and the probability ensemble method.

5.2.1. Dataset

Our extended experiments basically follow the MIREX 2009 setup. The evaluation data come from the MajorMiner's music labeling game[4], which invites players to listen to short music clips (about 10 seconds long) and label them with relevant words and phrases. According to the MIREX 2009 audio tag classification results web page, 45 tags, as listed in Table 4, are considered. We download all the audio clips that are associated with these 45 tags from the website of the MajorMiner's game. The resulting audio database contains 2,473 clips and the duration of each clip is 10 seconds or less. The dataset might be slightly different from that used in MIREX 2009 because the MajorMiner website might have been updated recently.

5.2.2. Model selection and evaluation

We evaluate our method with a three-fold cross-validation following the evaluation method in MIREX 2009. The 2,473 clips are split randomly into three subsets. In each fold, one subset is selected as the test set and the remaining two subsets serve as the training set. The test set for (outer) cross-validation is not used for determining the classifier setting. Instead, we first perform inner cross-validation on the held out data from the training set to determine the cost parameter C in the linear SVM and the number of base learners in AdaBoost. Then, we re-train the classifiers with the complete training set and the selected parameters, and perform outer cross-validation on the test set. Since the class distributions for some tags are imbalanced (more than two thousand negative instances and less than fifty positive instances), classification accuracy is not a fair criterion for model selection. Therefore, we use the AUC-ROC as the model selection criterion.

Table 4. The 45 tags used in the MIREX 2009 audio tag classification evaluation.

metal	instrumental	horns	piano	guitar
ambient	saxophone	house	loud	bass
fast	keyboard	electronic	noise	british
solo	electronica	beat	80s	dance
jazz	drum machine	strings	pop	r&b
female	rock	voice	rap	male
slow	vocal	quiet	techno	drum
funk	acoustic	distortion	organ	soft
country	hip hop	synth	trumpet	punk

To calculate the tag F-measure and tag accuracy, we need a threshold to binarize the output score. For the audio retrieval task, we want to retrieve audio clips from the audio database. We assume that each tag's class has similar probability distributions in the training and testing audio databases. Therefore, we set the threshold with the class prior distribution obtained from the training data. For the audio annotation task, we annotate the testing audio clips one by one. We set the threshold to 0.5 because the calibrated probability score ranges from 0 to 1.

5.2.3. Experiment results

Our experiment results in terms of the metrics corresponding to the audio retrieval task and the audio annotation task are summarized in Tables 5 and 6, respectively. Because the cross-validation split used in MIREX 2009 is not available, we perform three-fold cross-validation twenty times and calculate the mean and standard

[3] http://www.music-ir.org/mirex/2009/index.php/Audio_Tag_Classification

[4] http://majorminer.org/

deviation of the results to reduce the variance of different cross-validation splits.

Several observations can be drawn from Tables 5 and 6. First, pre-segmentation is effective. All the classification methods benefit from pre-segmentation. For example, the tag AUC-ROC is improved by 1.42% (cf. Linear SVM) and 4.23% (cf. AdaBoost). Second, SVM slightly outperforms AdaBoost. Third, the two ensemble methods are respectively suitable for either the retrieval task or the annotation task as discussed above. On the audio retrieval task, ranking ensemble not only has better mean performance than any individual classifier, but also has a smaller standard deviation. Probability ensemble is more suitable than ranking ensemble for the audio annotation task. However, the improvement over the SVM classifier is small.

Table 5. Audio retrieval results of different classifiers and ensemble methods on the MajorMiner dataset.

Mean± Standard Deviation	Tag AUC-ROC		Tag F-measure	
	Without Seg.	With Seg.	Without Seg.	With Seg.
AdaBoost	0.7520 ± 0.0026	0.7943 ±0.0024	0.2856 ± 0.0036	0.3034 ±0.0051
Linear SVM	0.7848 ± 0.0029	0.7990 ±0.0030	0.3092 ± 0.0028	0.3169 ±0.0038
Probability Ensemble	0.7894± 0.0030	0.8108± 0.0020	0.3163± 0.0037	0.3296± 0.0039
Ranking Ensemble	0.7997 ± 0.0022	**0.8189 ±0.0017**	0.3211 ± 0.0032	**0.3332 ±0.0038**

Table 6. Audio annotation results of different classifiers and ensemble methods on the MajorMiner dataset.

Mean± Standard Deviation	Clip AUC-ROC		Tag Accuracy	
	Without Seg.	With Seg.	Without Seg.	With Seg.
AdaBoost	0.8627 ± 0.0009	0.8774 ±0.0009	0.9162 ± 0.0004	0.9184 ±0.0004
Linear SVM	0.8788 ± 0.0009	0.8828 ±0.0012	0.9191 ± 0.0004	0.9200 ±0.0003
Probability Ensemble	0.8788 ± 0.0007	**0.8848 ±0.0007**	0.9191 ± 0.0002	**0.9201 ±0.0003**
Ranking Ensemble	0.7626± 0.0012	0.7814± 0.0010	0.9016 ± 0.0004	0.9057 ±0.0003

6. CONCLUSION

This paper presents our method for the audio tag annotation and retrieval task. Our major contributions include using the novelty curve to divide the audio clips into homogeneous segments and exploiting classifier ensemble. Our ranking ensemble method performs very well in the MIREX 2009 audio tag classification task in terms of tag AUC-ROC and tag F-measure but poorly in terms of clip AUC-ROC. Therefore, we have further proposed the probability ensemble method that performs very well in terms of clip AUC-ROC and tag accuracy. In other words, ranking ensemble is suitable for the audio retrieval task while probability ensemble is suitable for the audio annotation task.

Our future directions are as follows. First, we have realized that the audio tag classification task can be better formulated as a multi-label classification problem. Second, the frequency count of a tag can be taken into account in the training. For example, an audio clip that has been tagged many times should receive a larger weight for not being misclassified.

7. REFERENCES

[1] P. Lamere, "Social Tagging and Music Information Retrieval," *Journal of New Music Research*, vol. 37, no. 2, pp. 101–114, 2008.

[2] D. Eck, P. Lamere, T. Bertin-Mahieux, S. Green, "Automatic Generation of Social Tags for Music Recommendation", *NIPS*, 2007.

[3] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic Annotation and Retrieval of Music and Sound Effects," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 16, no. 2, pp. 467–476, 2008.

[4] M. I. Mandel and D. P. W. Ellis, "Multiple-Instance Learning for Music Information Retrieval," *ISMIR*, 2008.

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

[6] J. Foote and M. Cooper, "Media Segmentation using Self-Similarity Decomposition," *Proc. of SPIE Storage and Retrieval for Multimedia Databases*, vol. 5021, pp. 167-75, 2003.

[7] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, B. Kegl, "Aggregate Features and ADABOOST for Music Classification", *Machine Learning*, vol. 65, no. 2-3, pp. 473 – 484, 2006.

[8] O. Lartillot, P. Toiviainen and T. Eerola, *MIRtoolbox 1.1 User's Manual*, University of Jyväskylä, Finland, 2008.

[9] C. Duxbury, J. P. Bello, M. Davies, and M. Sandler, "Complex Domain Onset Detection for Musical Signals," *DAFx*, 2003.

[10] M. I. Mandel and D. P. W. Ellis, "Song-level features and support vector machines for music classification," *ISMIR*, 2005.

[11] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp.119–139, 1997.

[12] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods," *Advances in Large Margin Classifiers*, Cambridge, MA.

[13] H.-T. Lin, C.-J. Lin, and R. C. Weng, "A Note on Platt's Probabilistic Outputs for Support Vector Machines," *Machine Learning*, vol. 68, no.3, pp. 267-276, 2007.

[14] X. Hu, J. S. Downie, A. Ehmann, "Lyric Text Mining in Music Mood Classification," *ISMIR,* 2009.