Knowledge-Based Systems 125 (2017) 1–12

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Audio scene recognition based on audio events and topic model

Yan Leng[a], Nai Zhou[a], Chengli Sun[b], Xinyan Xu[c], Qi Yuan[a], Chuanfu Cheng[a], Yunxia Liu[d], Dengwang Li[a,*]

[a] Shandong Province Key Laboratory of Medical Physics and Image Processing Technology, Institute of Biomedical Sciences, School of Physics and Electronics, Shandong Normal University, Ji'nan 250014, China
[b] School of Information, Nanchang Hangkong University, Nanchang 330063, China
[c] Department of Computer Science and Technology, Shandong College of Electronic Technology, Ji'nan 250014, China
[d] Shandong Provincial Key Laboratory of Network Based Intelligent Computing, School of Information Science and Engineering, University of Jinan, Ji'nan 250014, China

ARTICLE INFO

Article history:
Received 9 May 2016
Revised 1 April 2017
Accepted 7 April 2017
Available online 8 April 2017

Keywords:
Audio scene recognition
Audio event
Topic model
PLSA
LDA
Support vector machine

ABSTRACT

Topic model is a hot research topic which is attracting attentions from many fields. Recently, several studies have applied topic model to ASR (audio scene recognition). Among these studies, most of them use the document-word co-occurrence matrix for topic analysis. In this work, we propose a new ASR algorithm based on audio events and topic model, which uses the document-event co-occurrence matrix for topic analysis. Our work is based on the hypothesis that: for an audio document, compared with its word distribution, its event distribution is more in line with humans' way of thinking, and then the topic distribution obtained based on the document-event co-occurrence matrix can represent the audio document better. The contribution of this work lies in that: (1) we propose an ASR algorithm which uses document-event co-occurrence matrix for topic analysis. Compared with the current studies which use document-word co-occurrence matrix for topic analysis, the proposed algorithm can extract the topic distribution which can express the audio documents better, and then can get better recognition results; (2) we propose a much easier method to obtain the document-event co-occurrence matrix; (3) we propose a method to weight the event distribution of audio documents; this weighting method can emphasize the audio events that are important in reflecting the unique topics of the audio documents, and can suppress the audio events that are common to many topics. Experimental results on two public datasets verify the effectiveness of the proposed ASR algorithm, and also verify the necessity and effectiveness of the proposed weighting method. The innovative ideas in this work are not limited to ASR, but can be extended to many other fields, such as the video classification etc.

1. Introduction

Audio scene recognition (ASR) refers to the task of identifying the environment for an audio stream in which it is produced, or in other words, it means using audio information to perceive the surrounding environment. Compared with vision information, audio information has many unique advantages: first, audio is not affected by light, audio-based system can work under weak light conditions; second, audio is not limited by the scope of vision, it can cover a wide range; third, audio can better protect people's privacy, and then can be applied in privacy occasions, such as bathroom and bedroom; besides, the acquisition cost of audio data is lower than that of vision data. Due to the above reasons, recently, audio information is widely used [1–3], and one of its important applications is ASR.

ASR has many useful applications. Applying ASR on the mobile devices can make the devices to be smart [4], for ASR enables them to perceive the surrounding environment, and then tune the status correspondingly; ASR can be applied to aquaculture industry [5], it enables the sound classifier to perform classification according to the context environment, and then can help to estimate the feed consumption of prawns more precisely; ASR can also be applied to smart home [6] etc.

In this work we want to introduce the topic model into ASR. Topic models have achieved great success in text analysis; recently, several studies have applied them to ASR. The paradigm of the methods used in these studies is similar to that of text analysis, the audio document is analogous to the text document, and the audio frames are analogous to the words. In that way, the common paradigm of ASR based on topic model consists of segmenting

* Corresponding author.
E-mail addresses: lyansdu@163.com, lidengwang@sdnu.edu.cn (D. Li).

http://dx.doi.org/10.1016/j.knosys.2017.04.001
0950-7051/

the audio documents into frames, creating the audio vocabulary by vector quantization, mapping the frames into audio words according to the audio vocabulary, counting the audio words to generate the document-word co-occurrence matrix, analyzing the co-occurrence matrix by the topic model to generate the topic distribution for each audio document, taking the topic distribution as the feature set to perform scene recognition. It can be seen that in these studies, document-word co-occurrence matrix is used for topic analysis, however, we think that using the document-event co-occurrence matrix for topic analysis would be more reasonable than using the document-word co-occurrence matrix, because the document-event co-occurrence matrix is more in line with humans' way of scene recognition. When we humans recognize the audio scene, we would first figure out what audio events are there in the audio document, and then by summing up the audio events we will think what are the topics that these audio events want to reflect; after analyzing the topics, we finally determine the type of audio scene. To this end, in this work, we hypothesize that: for an audio document, compared with its word distribution, its event distribution is more in line with humans' way of thinking, and then the topic distribution obtained based on the document-event co-occurrence matrix can represent the audio document better. Based on the above hypothesis, we propose an ASR algorithm which uses the document-event co-occurrence matrix for topic analysis.

The contribution of our work lies in that: (1) we propose an ASR algorithm which uses the document-event co-occurrence matrix for topic analysis. Compared with the current studies which use the document-word co-occurrence matrix for topic analysis, the proposed algorithm can extract the topic distribution which can express the audio documents better, and then can get better recognition results; (2) we propose a much easier method to get the document-event co-occurrence matrix. To obtain the document-event co-occurrence matrix, one natural way is to recognize the audio events in the audio documents through classification model first, and then perform statistical analysis, but this method needs to construct the classification model, when the number of audio events is large, the amount of calculation will be great, while our proposed method does not need to construct the classification model, but only needs to do a matrix factorization through the topic model. Besides, another problem for obtaining the document-event co-occurrence matrix through classification model is that: due to the misclassification of audio events, for the same audio scene class, its document-event co-occurrence matrix obtained from the test set may have poor consistency with that obtained from the training set, while the proposed method would avoid this problem; (3) we propose a method to weight the event distribution of audio documents. This weighting method would emphasize the audio events that play important roles in reflecting the unique topics of the audio documents, and would suppress the audio events that are common to many topics. As a result, it can help to extract the topic distribution that can better express the audio documents. Experimental results on two public datasets have verified the effectiveness of the proposed ASR algorithm, and have verified the necessity and the effectiveness of the proposed weighting method.

The rest of the paper is organized as follows. Section 2 discusses the related work; Section 3 briefly introduces two topic models: PLSA and LDA; Section 4 describes the proposed ASR algorithm; Section 5 shows the experimental results, and Section 6 gives conclusions and future work.

2. Related work

Many kinds of topic models have been proposed for text analysis, among them, PLSA (Probabilistic Latent Semantic Analysis) and LDA (Latent Dirichlet Allocation) are the two most popular ones. PLSA was first proposed by Hofmann [7], it models each document as a distribution over latent topics, and models each topic as a distribution over words, but it does not make any assumption about the generation of the document-topic distribution. Later, Blei et al. [8] extended PLSA by introducing a Dirichlet prior on the document-topic distribution, and proposed LDA.

PLSA and LDA have achieved great success in text analysis. Hofmann applied PLSA to automated indexing of documents [7]; Xu et al. used LDA to identify the implicit feature in Chinese reviews [9], and Zhang et al. utilized LDA to improve short text classification [10]. PLSA and LDA are not limited to text analysis; they have also been applied to many other fields. For example, Pliakos et al. applied PLSA to image classification [11]; Zhou et al. applied LDA to expert finding in question answer communities [12]. PLSA and LDA have also been applied to audio field [13–20]. Hazen et al. [13] used PLSA to summarize the topic content of the audio corpus. In [13], for each audio document, the occurrence number of each word was first estimated through automatic speech recognition system; then PLSA was used to learn the latent topics of the audio documents; these latent topics were then ranked according to importance; finally, signature words were adopted to describe the content of the topics. Mesaros et al. [14] used PLSA to help to detect audio events. In [14], HMM was used to detect the audio events; in order to generate the inter-model transition probabilities of the HMM network, PLSA was adopted to estimate the prior probabilities of audio events, and these prior probabilities were then used as the inter-model transition probabilities. Hu et al. [15] and Kim et al. [16] applied LDA to audio retrieval. In [15], the authors improved the traditional LDA, and proposed Gaussian-LDA for audio retrieval. Different from the traditional LDA which uses multinomial distribution to model the topic-word distribution, Gaussian-LDA adopts Gaussian distribution to model the topic-word distribution. In this way, Gaussian-LDA can avoid information loss caused by VQ (vector quantization). In [16], the authors created the audio vocabulary though LBG-VQ (Linde–Buzo–Gray Vector Quantization), extracted the topic distribution of each audio clip through LDA, and finally adopted SVM (Support Vector Machines) to do classification. Later in 2012, the algorithm proposed in [16] was applied to audio tag classification [17].

There are also studies which applied PLSA and LDA to ASR [18–20]. All these methods follow the common paradigm of ASR which is described in the introduction section, including: creating audio vocabulary, mapping audio frames into audio words, counting document-word co-occurrence matrix, generating the document-topic distribution for each audio clip through topic model and performing classification. In [18] and [19], the authors adopted PLSA as the topic model, utilized SVM to do classification, and used RPCL (Rival Penalized Competitive Learning) clustering and GMM (Gaussian Mixture Model) clustering to create the audio vocabulary respectively. In [20], Kim et al. adopted LDA as the topic model, utilized SVM to do classification, and used LBG-VQ to create the audio vocabulary.

In this work we also adopt PLSA and LDA to perform ASR. Compared with the above PLSA/LDA based ASR algorithms [18–20], the difference of our proposed algorithm is that it uses the document-event co-occurrence matrix for topic analysis, while in [18–20], the authors used the document-word co-occurrence matrix for topic analysis. Our work of using the document-event co-occurrence matrix for topic analysis is based on the hypothesis that: for an audio document, compared with its word distribution, its event distribution is more in line with humans' way of thinking, and then the topic distribution obtained based on the document-event co-occurrence matrix can represent the audio document better.

3. PLSA and LDA

3.1. PLSA

PLSA is a generative model which was first proposed by Hofmann for text analysis [7]. Let D denote a text collection containing N documents, $D = \{d_1, \cdots, d_N\}$; let W denote the vocabulary containing all different words occurring in D, $W = \{w_1, \cdots, w_V\}$; assuming there are T latent topics, denoted as $Z = \{z_1, \cdots, z_T\}$, and each word in a document is associated with a topic, then the generation of the text collection can be modeled as follows:

1. Choose a document d_i with probability $p(d)$;
2. Choose a topic z_j with probability $p(z|d_i)$;
3. Choose a word w_k with probability $p(w|z_j)$;
4. Repeat the above process to generate the whole text collection.

$p(d)$ denotes the probability distribution over documents; $p(z|d_i)$ denotes the probability distribution over topics given document d_i; $p(w|z_j)$ denotes the probability distribution over words given topic z_j.

Let matrix Θ denote the multinomial distribution of documents over topics, each column corresponds to one document, which is to say that its ith column $\Theta^{(d_i)} = P(z|d_i)$ refers to the multinomial distribution of document d_i over topics; similarly, let matrix Φ denote the multinomial distribution of topics over words, each column corresponds to one topic, which is to say that its jth column $\Phi^{(z_j)} = P(w|z_j)$ refers to the multinomial distribution of topic z_j over words. The key point of PLSA is to obtain Θ and Φ, and this can be achieved through maximizing the likelihood function by EM (expectation-maximization) as follows:

E-step:

$$p(z_j|w_k, d_i) = \frac{p(w_k|z_j)\,p(z_j|d_i)}{\sum_{m=1}^{T} p(w_k|z_m)\,p(z_m|d_i)} \tag{1}$$

M-step:

$$p(w_k|z_j) = \frac{\sum_{i=1}^{N} n(w_k, d_i)\,p(z_j|w_k, d_i)}{\sum_{m=1}^{V}\sum_{i=1}^{N} n(w_m, d_i)\,p(z_j|w_m, d_i)} \tag{2}$$

$$p(z_j|d_i) = \frac{\sum_{k=1}^{V} n(w_k, d_i)\,p(z_j|w_k, d_i)}{\sum_{m=1}^{T}\sum_{k=1}^{V} n(w_k, d_i)\,p(z_m|w_k, d_i)} \tag{3}$$

$n(w_k, d_i)$ denotes the occurrence number of word w_k in document d_i. In the test stage, when a new document d_{new} is given, its topic distribution $p(z|d_{new})$ can be estimated in a way similar to that used in the training stage, but should keep $P(w|z_j)$ $(j = 1, \cdots, T)$ learned in the training stage fixed:

E-step:

$$p(z_j|w_k, d_{new}) = \frac{p(w_k|z_j)\,p(z_j|d_{new})}{\sum_{m=1}^{T} p(w_k|z_m)\,p(z_m|d_{new})} \tag{4}$$

M-step:

$$p(z_j|d_{new}) = \frac{\sum_{k=1}^{V} n(w_k, d_{new})\,p(z_j|w_k, d_{new})}{\sum_{m=1}^{T}\sum_{k=1}^{V} n(w_k, d_{new})\,p(z_m|w_k, d_{new})} \tag{5}$$

3.2. LDA

LDA is a generative model which was first proposed by Blei et al. for text analysis [8]. Compared with PLSA, LDA introduces a Dirichlet prior for Θ. In [8], Blei et al. also discussed a smoothed LDA which introduces a Dirichlet prior for Φ as well. Let α and β denote the hyperparameters specifying the nature of the prior on Θ and Φ respectively, then the complete probability model is:

$$\begin{aligned} \Theta &\sim Dirichlet(\alpha) \\ z\big|\Theta^{(d_i)} &\sim Multinomial(\Theta^{(d_i)}) \\ \Phi &\sim Dirichlet(\beta), \\ w\big|\Phi^{(z_j)} &\sim Multinomial(\Phi^{(z_j)}) \end{aligned} \tag{6}$$

and the generation of the text collection can be modeled as follows:

1. For each document d_i, choose $\Theta^{(d_i)} \sim Dirichlet(\alpha)$;
2. For each word w_k in document d_i:
   (a) Choose a topic $z_j \sim Multinomial(\Theta^{(d_i)})$;
   (b) Choose $\Phi^{(z_j)} \sim Dirichlet(\beta)$
   (c) Choose a word $w_k \sim Multinomial(\Phi^{(z_j)})$

Many methods have been proposed to learn LDA, and the two most popular ones are Gibbs sampling and variational inference. For our proposed ASR algorithm, since the document-event co-occurrence matrix is obtained through matrix factorization, and we can not know the document index for each occurrence of an audio event, then Gibbs sampling is not applicable, to this end, we adopt variational inference for LDA learning.

PLSA and LDA are two different kinds of topic model, as that pointed out in [21], they can be interpreted as matrix factorization, just as that shown in Fig. 1. In Fig. 1, C denotes the document-word co-occurrence matrix; after the matrix factorization of the topic model, it is decomposed into two parts: the topic-word distribution matrix Φ and the document-topic distribution matrix Θ.

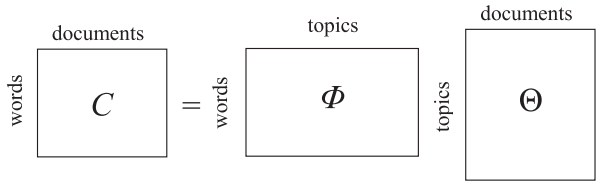


Fig. 1. The matrix factorization of the topic model.

4. The proposed ASR algorithm

The framework of the proposed ASR algorithm is shown in Fig. 2. It mainly consists of three parts: creating audio vocabulary, generating document-event co-occurrence matrix and performing audio scene recognition based on topic model.

4.1. Creating audio vocabulary

In this work we segment the audio documents into frames of 30 ms long using the hamming window with 50% overlap. For each frame, 39-dimensional MFCCs (Mel Frequency Cepstrum Coefficients) are extracted. One kind of clustering algorithm is then used to cluster the frames to create the audio vocabulary. Here, the clustering is not limited to a certain kind of clustering algorithm, because for different training set, different clustering algorithm may be suitable. The optimal number of clusters is determined by the method used in [22] where several values of clustering number were tried, and the final clustering number was determined by evaluating the quality of each resulting clustering using the BIC (Bayesian Information Criteria) criterion. Considering that the number of frames in the training set would be too large to be clustered for the reason of running out of memory, here we perform clustering on each audio scene class. For each audio scene class of the training set, the frames are clustered, and the cluster centroids are collected; the cluster centroids collected from all audio scene classes constitute the audio vocabulary.

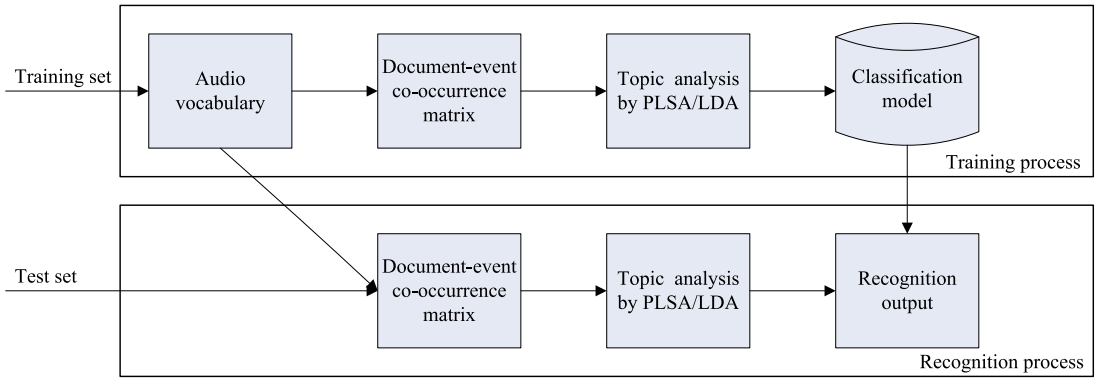


Fig. 2. The framework of the proposed ASR algorithm.

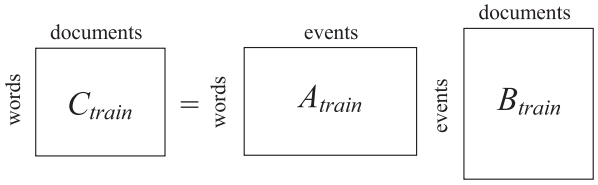


Fig. 3. Obtaining the document-event co-occurrence matrix B_{train} through decomposing C_{train} by PLSA. C_{train} and A_{train} are obtained through counting the document-word co-occurrence times and the event-word co-occurrence times for the training set respectively.

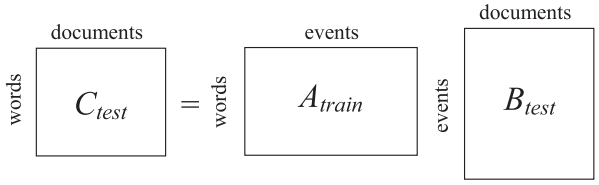


Fig. 4. Obtaining the document-event co-occurrence matrix B_{test} through decomposing C_{test} by PLSA. C_{test} is the document-word co-occurrence matrix of the test set obtained through counting, and A_{train} is the event-word co-occurrence matrix of the training set obtained through counting.

4.2. Generating document-event co-occurrence matrix

With the audio vocabulary, each audio frame in the training set is then mapped into audio word by choosing the closest word in the vocabulary. After that, we count the document-word co-occurrence matrix for the training set, denoted as C_{train}. Since in the training set, the audio event labels for the audio frames can be known in advance, we can also count the event-word co-occurrence matrix, denoted as A_{train}. Assuming there are N documents $\{d_1, \cdots, d_N\}$ and n audio events $\{e_1, e_2, \cdots, e_n\}$ in the training set, and the size of the audio vocabulary is M, then C_{train} is an $M \times N$ matrix, and A_{train} is an $M \times n$ matrix. Let B_{train} denote the document-event co-occurrence matrix for the training set, B_{train} is a matrix taking the form of $[p_{e_i}^{d_j}]_{n \times N}$, each column corresponds to one audio document. $p_{e_i}^{d_j}$ is the (i,j)th element of B_{train}, it represents the distribution of document d_j on event e_i. B_{train} can be obtained through decomposing C_{train} by PLSA with A_{train} fixed, just as that shown in Fig. 3.

When counting the event-word co-occurrence matrix A_{train}, it should be noticed that in the audio documents there is always the case that several audio events happen simultaneously in time. In this case, when performing annotation, for the time interval which contains multiple audio events, we annotate as many events contained in it as possible, but no more than 3, because we find that most people can not perceive more than 3 audio events at the same time. Under this situation, when counting the event-word co-occurrence matrix, the frame which has multiple labels will participate in the statistics of all audio events contained in it with uniform proportions. For example, if m audio events are labeled for a frame, then for each of the m audio events, when counting the event-word co-occurrence times, this frame will contribute $1/m$ to it.

It needs to be pointed out that for the time intervals which contain multiple audio events, it is very likely that different annotator will produce different annotation result. In order to reduce the annotation inconsistency between different annotators, in this work, we invite 3 annotators to annotate the same document; for a time interval which contains multiple audio events, in its annotation result, if an event label is annotated by two or more annotators, then this event label is retained, otherwise it is neglected.

In the test stage, after segmenting the test audio documents into frames and mapping the frames into audio words, we could count the document-word co-occurrence matrix for the test set, denoted as C_{test}. Keeping the event-word co-occurrence matrix used in the test stage the same as the event-word co-occurrence matrix obtained in the training stage, that is A_{train}, we can then obtain the document-event co-occurrence matrix for the test set (denoted as B_{test}) in a way similar to that used in the training stage, just as that shown in Fig. 4.

In the above descriptions, both the document-event co-occurrence matrix of the training set (that is B_{train}) and the document-event co-occurrence matrix of the test set (that is B_{test}) are obtained through PLSA matrix factorization. It should be noticed that there is actually another scheme to obtain B_{train} and B_{test}: in the training set, the audio event labels of the audio frames can be known in advance, then B_{train} can be obtained through counting; through PLSA matrix factorization, C_{train} can be decomposed into two parts: A_{train} and B_{train}; with A_{train} fixed, C_{test} can be decomposed into two parts: A_{train} and B_{test}. In this scheme, B_{train} is obtained through counting, while B_{test} is obtained through matrix factorization, so B_{train} and B_{test} are obtained through different ways; the different obtaining way would inevitably cause certain inconsistency between B_{train} and B_{test}, and this inconsistency would then affect the recognition results. In the "Experimental results" section we have done experiments to make a comparison between these two schemes, and the experimental results validate that our proposed scheme is more effective. Therefore, in this work the latter scheme is not adopted.

The importance of the events in helping to recognize the audio scene is different, and then in order to emphasize the events which are more important for recognition, it is necessary to weight the event distribution of the audio documents. In this work, the topic

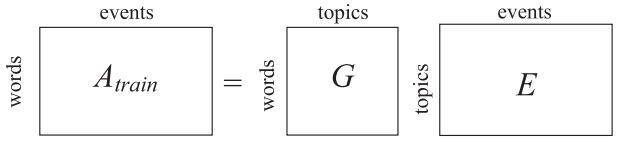


Fig. 5. Obtaining the event-topic distribution matrix E through decomposing A_{train} by PLSA. After matrix factorization, A_{train} is decomposed into two parts: G and E. A_{train} is the event-word co-occurrence matrix of the training set obtained through counting.

distribution is used as the feature set of the audio documents, and is used as the input of the classification model, then we think that the events whose topic is prominent, or in other words, the events whose occurrence would only reflect few topics are more important, and the events that are common to many topics are less important. Here we will use entropy to reflect the importance of the events. To this end, we first decompose the event-word co-occurrence matrix A_{train} through PLSA, just as that shown in Fig. 5. Assuming there are $T1$ latent topics $\{z_1, \cdots, z_{T1}\}$, then after matrix factorization, we would get the event-topic distribution matrix E taking the form of $[p_{z_i}^{e_j}]_{T1\times n}$, $(i = 1, \cdots, T1,\ j = 1, \cdots, n)$. $p_{z_i}^{e_j}$ represents the distribution of event e_j on topic z_i. With the event-topic distributions, the entropy of the events can then be calculated. We use vector $H = [H(e_j)]_{1\times n}$ to denote the entropy values of the events. $H(e_j)$ represents the entropy value of event e_j which can be calculated as follows:

$$H(e_j) = -\sum_{i=1}^{T1} p_{z_i}^{e_j} \log_2(p_{z_i}^{e_j}) \tag{7}$$

For an audio event, small entropy value means that its topic is very prominent, or in other words, it means that its occurrence would only reflect few topics; large entropy value means that this audio event is common to many topics. Therefore, the audio event which has a smaller entropy value is more important for recognition.

Based on entropy, here we design a coefficient to evaluate the importance of the audio event, and use it to weight the event distribution of the audio documents. The coefficient is designed based on the following two criteria: first, the event which has a smaller entropy value should obtain a larger coefficient to reflect its importance; second, since $p_{e_i}^{d_j}$ takes values in [0,1], in order to prevent the case that after weighting, some weighted $p_{e_i}^{d_j}$ values would be too small, we restrict that the coefficient should be greater than or equal to 1. We use vector c to represent the coefficients of the events, $c(e_i)$ denotes the coefficient of event e_i which is designed as follows:

$$c(e_i) = e^{-|H(e_i)-\min(H)|/2\mathrm{var}(H)} \tag{8}$$

$$c(e_i) \Leftarrow c(e_i)/\min(c) \tag{9}$$

$\min(\cdot)$ and $\mathrm{var}(\cdot)$ mean to calculate the minimum value and the variance of the elements in a vector respectively. Formula (8) can ensure that a smaller entropy value will get a larger coefficient, and formula (9) can ensure the coefficient to be greater than or equal to 1. Coefficient vector c is then used to weight the document-event distribution in B_{train} and B_{test} respectively. Taking B_{train} as the example, the update formula of the document-event distribution is:

$$p_{e_i}^{d_j} \Leftarrow c(e_i) \cdot p_{e_i}^{d_j} \quad (i = 1, \cdots, n;\ j = 1, \cdots, N) \tag{10}$$

When performing topic analysis by PLSA or LDA, it is hoped that each column of B_{train} and B_{test} is the occurrence number of the events in an audio document, while currently each column of B_{train} and B_{test} is the weighted probability distribution of an audio document over audio events, to this end, we adjust each column of B_{train} and B_{test} as follows:

$$p^{d_j} \Leftarrow norm(p^{d_j}) \tag{11}$$

$$p^{d_j} \Leftarrow fNum \cdot p^{d_j} \tag{12}$$

p^{d_j} represents the event distribution of document d_j in B_{train} or B_{test}; $norm(\cdot)$ represents normalization operation; $fNum$ represents the total frame number of the training set or test set. The updated B_{train} and B_{test} can then represent the document-event co-occurrence matrix of the training set and the test set respectively.

4.3. Audio scene recognition based on topic model

As that discussed in the introduction section, when performing ASR based on topic model, the topic distribution is used as the feature set of the audio documents, and is used as the input of the classification model. To obtain the topic distribution of the training audio documents and the test audio documents, B_{train} and B_{test} are decomposed through the topic model respectively. As that shown in Fig. 1, through the topic model, B_{train} can be decomposed into two parts: P_{train} and Q_{train} which are equivalent to matrix Φ and Θ in Fig. 1 respectively; taking P_{train} fixed, B_{test} can be decomposed into two parts: P_{train} and Q_{test}. Assuming there are $T2$ latent topics, then Q_{train} is a $T2 \times N$ matrix whose each column represents the topic distribution of a training audio document; assuming there are N_{test} audio documents in the test set, then Q_{test} is a $T2 \times N_{test}$ matrix whose each column represents the topic distribution of a test audio document.

Taking the topic distribution as the feature set of the audio documents, we then use SVM as the classification model to perform ASR. SVM is a binary classifier which has been widely used in audio field [1,18–20,23], when classifying multiple classes of audio scene, in order to avoid the data unbalance problem, here we adopt the one-vs-one multiclass classification strategy.

5. Experimental results

5.1. Dataset

We test the proposed algorithm on two public datasets. The first one is the dataset for the scene classification task in IEEE AASP challenge [24]. This dataset consists of 2 equally proportioned parts, one is publicly released as a development set, and the other is held-back for evaluating submissions. In this work we use its development set for experiments, and denote it as AASP dataset. There are totally 10 classes of audio scene in AASP dataset, including busy street, quiet street, supermarket, restaurant, office, park, bus, tube, tube station and open market. Each audio scene class contains 10 audio documents, each of which is 30 s long, 16 bit, stereo and sampled in 44.1 k Hz. The audio documents are transformed into mono channel format, and down-sampled to 16 k Hz. The other dataset is the DEMAND (Diverse Environments Multichannel Acoustic Noise Database) dataset [25] which provides a set of 16-channel audio files recorded in a variety of indoor and outdoor settings. There are totally 18 audio scene classes in DEMAND, including kitchen, living, washing, field, park, river, hallway, meeting, office, cafeter, restaurant, station, cafe, square, traffic, bus, car and metro. Each audio scene class contains 16 recordings corresponding to 16 channels. In this work we only use the first channel recording for experiments. Each recording is 300 s long and sampled in 16 k Hz, it is then segmented equally into 10 audio documents, each 30 s long. In summary, in DEMAND dataset there are 18 audio scene classes, each class contains 10 audio documents of 30 s long.

5.2. Experimental setting and results

As that described in Section 4, audio documents are segmented into 30 ms-long frames using the hamming window with 50% overlap; for each frame, 39-dimensional MFCCs are extracted as the feature set; after performing topic analysis through PLSA/LDA, the topic distribution is used to represent each audio document, and is taken as the input of SVM. For SVM, one-vs-one strategy is adopted for multiclass classification; RBF (Radial Basis Function) is adopted as the kernel function; the penalty coefficient and the gamma parameter of the kernel function are determined through the grid search method.

Our proposed algorithm uses the document-event (DE) co-occurrence matrix for topic analysis, and adopts PLSA and LDA as the topic model, so hereafter we will use DE_PLSA to represent the proposed algorithm which adopts PLSA as the topic model, and use DE_LDA to represent the proposed algorithm which adopts LDA as the topic model. The algorithm proposed in [19] used the document-word (DW) co-occurrence matrix for topic analysis, and adopted PLSA as the topic model, so hereafter we will denote it as DW_PLSA. The algorithm proposed in [20] used the document-word co-occurrence matrix for topic analysis, and adopted LDA as the topic model, so hereafter we will denote it as DW_LDA.

There are totally three adjustable parameters in the proposed ASR algorithm: M, the size of the audio vocabulary; $T1$, the number of topics in the event-topic distribution of matrix E; $T2$, the number of topics in the document-topic distribution of matrix Q_{train} and Q_{test}. In order to determine M, when performing clustering on each audio scene class in the training stage, the clustering number of each audio scene class is set to be 50, 100, 150, 200, 250 and 300 respectively, and then as that done in [22], the quality of each resulting clustering is evaluated through the BIC criterion, finally, the optimal number of clusters is determined as the one with the best quality of clustering. For AASP dataset, the clustering number of each class is determined to be 150, and then $M = 150 \times 10 = 1500$; for DEMAND dataset, the clustering number of each class is determined to be 300, and then $M = 300 \times 18 = 5400$. The method of determining $T1$ and $T2$ is as follows: the proposed algorithm is evaluated through 5-fold cross validation. In the first fold, the first two audio documents of each class are used for testing, and the others for training; in the next fold, the next two audio documents of each class are used for testing, and the others for training, and so on. In each fold, $T1$ and $T2$ are set to be different values respectively; under each pair of ($T1$,$T2$), K-fold cross validation is performed on the training audio documents, specifically, the training audio documents are equally divided into K parts, the (K-1) parts alternate as the training set, and the other 1 part as the validation set; under each pair of ($T1$,$T2$), the performance of the proposed algorithm is obtained through averaging the performance of these K-fold. This average performance is simply the performance of the proposed algorithm in one of the 5-fold cross validation; under each pair of ($T1$,$T2$), we further average the performance coming from the 5-fold to obtain the final performance. The optimal $T1$ and $T2$ are determined as the pair of ($T1$,$T2$) under which the proposed algorithm gets the best performance.

The algorithms are evaluated in terms of classification accuracy, standard deviation and confusion matrix etc. through 5-fold cross validation. As that stated in the previous paragraph, in the first fold, the first two audio documents of each class are used for testing, and the others for training; in the next fold, the next two audio documents of each class are used for testing, and the others for training, and so on. The final results are obtained by summing up the results of each fold. The classification accuracy is defined as follows:

$$accuracy = \frac{the\ number\ of\ correctly\ classified\ audio\ documents}{the\ total\ number\ of\ audio\ documents\ in\ the\ test\ set} \tag{13}$$

5.2.1. Determination of T1 and T2

During experiments we found that under larger values of $T1$ and $T2$, the classification performance of the proposed algorithm is not better than that under smaller values of $T1$ and $T2$, moreover, considering that smaller values of $T1$ and $T2$ mean less amount of calculation, so here we search the optimal $T1$ and $T2$ from small values. As that discussed in the above section, we set $T1$, $T2 = 10$, 20, 30, …, 100 respectively; under each pair of ($T1$,$T2$), the final performance of the proposed algorithm is obtained through averaging the performance coming from the 5-fold cross validation. Under different pairs of ($T1$,$T2$), the classification accuracy of the proposed ASR algorithm on AASP and DEMAND dataset is shown in Figs. 6 and 7 respectively. In each figure, subfigure (a) shows the classification accuracy of DE_PLSA, and subfigure (b) shows the classification accuracy of DE_LDA. As that discussed in the next section, for DE_PLSA, GMM clustering has been adopted to create the audio vocabulary, and for DE_LDA, LBG-VQ clustering has been adopted to create the audio vocabulary.

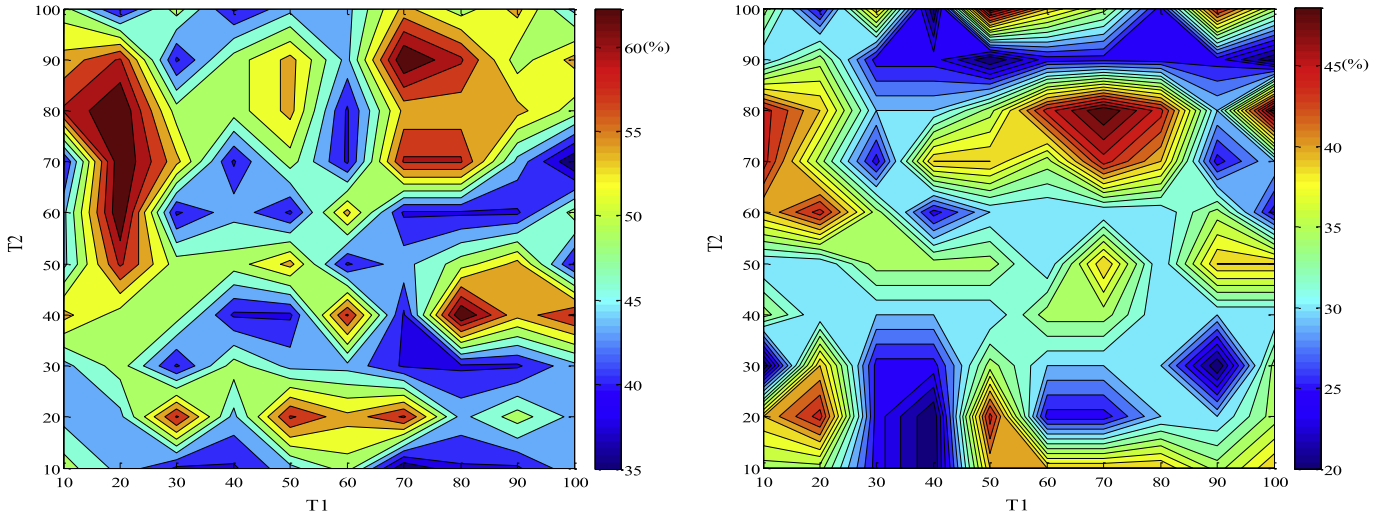
From Figs. 6 and 7 it can be seen that on both datasets, the proposed ASR algorithm can get better performance under smaller values of $T1$, $T2$, then on AASP dataset, we set $T1 = 20$, $T2 = 60$ for DE_PLSA, and set $T1 = 70$, $T2 = 80$ for DE_LDA; on DEMAND dataset, we set $T1 = 20$, $T2 = 10$ for DE_PLSA, and set $T1 = 30$, $T2 = 10$ for DE_LDA.

5.2.2. The effectiveness of the proposed ASR algorithm

To verify that the proposed algorithm which uses the document-event co-occurrence matrix for topic analysis is more effective than the current algorithms which use the document-word co-occurrence matrix for topic analysis, here we compare DE_PLSA with DW_PLSA, and compare DE_LDA with DW_LDA. As that discussed in Section 3.2, for the proposed DE_LDA, Gibbs sampling is not suitable for LDA learning, and then we adopt variational inference. In that case, we use variational inference for the LDA learning of DW_LDA as well. The DW_PLSA algorithm proposed in [19] and the DW_LDA algorithm proposed in [20] use GMM clustering and LBG-VQ clustering to create the audio vocabulary respectively, while our proposed algorithm does not limited to a certain kind of clustering algorithm, therefore, to be fair, when compared with DW_PLSA, DE_PLSA adopts the clustering algorithm used in DW_PLSA, i.e. GMM clustering, and when compared with DW_LDA, DE_LDA adopts the clustering algorithm used in DW_LDA, i.e. LBG-VQ clustering. Both DW_PLSA and DW_LDA have two adjustable parameters: M, the size of the audio vocabulary; $T2$, the number of topics in the document-topic distribution. To be fair in comparison, M is set to the same value as that of DE_PLSA/DE_LDA, and $T2$ is determined in the same way as that used in DE_PLSA/DE_LDA.

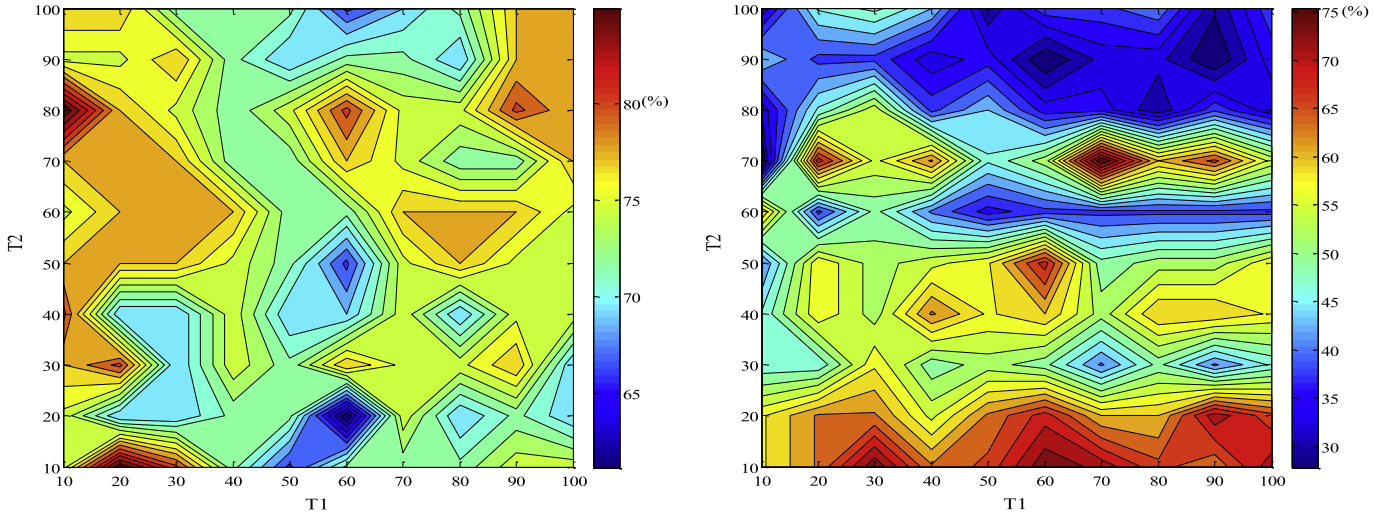
For DW_PLSA, DE_PLSA, DW_LDA and DE_LDA, their classification accuracy and standard deviation on AASP and DEMAND dataset are shown in Table 1; their confusion matrices on AASP and DEMAND dataset are shown in Table 2 and Table 3 respectively. On AASP dataset, 1 = bus, 2 = busystreet, 3 = office, 4 = openairmarket, 5 = park, 6 = quietstreet, 7 = restaurant, 8 = supermarket, 9 = tube, 10 = tubestation. On DEMAND dataset, 1 = kitchen, 2 = living, 3 = washing, 4 = field, 5 = park, 6 = river, 7 = hallway, 8 = meeting, 9 = office, 10 = cafeter, 11 = restaurant, 12 = station, 13 = café, 14 = square, 15 = traffic, 16 = bus, 17 = car, 18 = metro.

From Table 1 it can be seen that on both datasets, by the usage of document-event co-occurrence matrix, the proposed DE_PLSA performs much better than DW_PLSA, and the proposed DE_LDA



(a) DE_PLSA (b) DE_LDA

Fig. 6. The classification accuracy of the proposed ASR algorithm under different pairs of (*T*1,*T*2) on AASP dataset. Subfigure (a) is the classification accuracy of DE_PLSA, and subfigure (b) is the classification accuracy of DE_LDA.



(a) DE_PLSA (b) DE_LDA

Fig. 7. The classification accuracy of the proposed ASR algorithm under different pairs of (*T*1,*T*2) on DEMAND dataset. Subfigure (a) is the classification accuracy of DE_PLSA, and subfigure (b) is the classification accuracy of DE_LDA.

Table 1
The classification performance of DW_PLSA, DE_PLSA, DW_LDA and DE_LDA on AASP and DEMAND dataset.

Dataset	Topic model	Algorithm	Accuracy (%)	St. dev.
AASP	PLSA	DW_PLSA	46.0	5.5
		DE_PLSA	**61.0**	6.5
	LDA	DW_LDA	48.0	4.5
		DE_LDA	**54.0**	2.2
DEMAND	PLSA	DW_PLSA	63.3	6.0
		DE_PLSA	**83.3**	3.9
	LDA	DW_LDA	66.7	10.9
		DE_LDA	**78.9**	9.7

performs much better than DW_LDA, which verifies the correctness of the proposed idea that using the document-event co-occurrence matrix for topic analysis is more reasonable than using the document-word co-occurrence matrix. The reason may be that using the document-event co-occurrence matrix is more in line with humans' way of scene recognition, and then the extracted document-topic distribution can express the audio documents better.

From Table 2 it can be seen that on AASP dataset, DE_PLSA performs much better than DW_PLSA in recognizing the audio scene classes of bus, quietstreet and tubestation, and DE_LDA performs much better than DW_LDA in recognizing the audio scene class of bus. Both DE_PLSA and DE_LDA have a good performance in recognizing the audio scene classes of bus, busystreet, office, openairmarket and quietstreet, but they perform badly in recognizing park, restaurant, supermarket and tube. The reason for the good performance on bus, busystreet, office, openairmarket and quietstreet may be that: for each of these 5 audio scene classes, the audio events which are important in reflecting the topics of the audio documents occur in most of the 10 audio documents, and then after the topic analysis, the topic distribution of the training documents and the topic distribution of the test documents are more

Table 2
The confusion matrices of DW_PLSA, DE_PLSA, DW_LDA and DE_LDA on AASP dataset.

Actual Classes / Predicted Classes

Predicted \ Actual	1	2	3	4	5	6	7	8	9	10
1	5	0	0	0	0	0	0	0	0	0
2	2	9	0	0	0	0	0	2	3	0
3	0	0	10	0	1	1	0	1	1	0
4	0	0	0	6	0	1	1	0	0	0
5	0	0	0	3	6	1	1	1	0	0
6	0	0	0	0	0	3	1	0	0	0
7	1	0	0	1	0	2	2	1	1	2
8	1	1	0	0	1	2	4	1	2	0
9	1	0	0	0	1	0	1	4	3	7
10	0	0	0	0	1	0	0	0	0	1

(a) DW_PLSA

Actual Classes / Predicted Classes

Predicted \ Actual	1	2	3	4	5	6	7	8	9	10
1	10	0	0	0	1	0	0	1	1	1
2	0	10	0	0	0	0	0	0	0	0
3	0	0	10	0	1	0	0	0	0	0
4	0	0	0	9	0	1	6	1	1	1
5	0	0	0	1	3	1	0	0	0	0
6	0	0	0	0	5	8	0	1	1	0
7	0	0	0	0	0	0	1	1	0	0
8	0	0	0	0	0	0	0	1	0	1
9	0	0	0	0	0	0	0	2	2	0
10	0	0	0	0	0	0	3	3	5	7

(b) DE_PLSA

Actual Classes / Predicted Classes

Predicted \ Actual	1	2	3	4	5	6	7	8	9	10
1	4	0	0	0	0	0	0	0	0	0
2	2	8	0	2	0	0	0	1	0	1
3	0	0	10	0	3	1	0	1	1	0
4	0	0	0	6	1	0	2	2	3	2
5	0	0	0	0	2	1	1	0	0	0
6	1	0	0	1	0	8	2	1	0	0
7	3	1	0	0	1	0	3	2	1	2
8	0	1	0	0	1	0	0	1	2	0
9	0	0	0	1	2	0	1	1	3	2
10	0	0	0	0	0	0	1	1	0	3

(c) DW_LDA

Actual Classes / Predicted Classes

Predicted \ Actual	1	2	3	4	5	6	7	8	9	10
1	10	1	0	0	0	0	0	0	0	0
2	0	8	0	0	0	1	3	1	2	1
3	0	0	10	0	3	0	0	1	0	1
4	0	0	0	9	2	2	2	3	1	3
5	0	0	0	0	2	0	0	1	1	2
6	0	0	0	1	1	7	0	0	0	0
7	0	0	0	0	0	0	2	1	0	0
8	0	1	0	0	1	0	3	2	0	0
9	0	0	0	0	1	0	0	0	1	0
10	0	0	0	0	0	0	0	1	5	3

(d) DE_LDA

consistent. For example, for the bus class, in all of the 10 audio documents, there are the audio events of bus sound and speech which are important in reflecting the topics of the bus scene. The reasons for the bad performance on park, restaurant, supermarket and tube may be that: for the park class, the typical audio events, such as the chirping of birds, occur only in less than 50% of the total audio documents, and then the topic distribution of the training documents and the topic distribution of the test documents would have a poor consistency; for restaurant and supermarket, in all of the 10 audio documents, there are nearly no typical audio events,

Table 3
The confusion matrices of DW_PLSA, DE_PLSA, DW_LDA and DE_LDA on DEMAND dataset.

Actual Classes

Predicted Classes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	10	0	1	2	3	0	1	0	3	0	0	0	0	0	0	0	0	0
2	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	2	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	6	0	3	2	3	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	10	0	0	0	0	0	0	0	2	0	2	4	0
7	0	1	0	2	0	0	3	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0	4	0	0	1	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	6	0	0	4	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
13	0	0	0	0	1	0	2	0	0	4	0	0	1	2	0	0	0	0
14	0	0	0	0	0	0	1	0	0	0	0	0	1	3	0	1	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	2	3	0	7	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0
18	0	0	0	0	0	0	0	0	0	0	3	7	2	0	0	0	0	10

(a) DW_PLSA

Actual Classes

Predicted Classes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	2	0	0	10	0	2	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	5	0	0	1	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	2	0	10	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0
11	0	1	0	0	0	0	0	0	0	0	10	0	1	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
13	0	1	0	0	0	0	1	1	0	0	0	0	9	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0
18	0	1	0	0	0	0	0	0	0	0	0	8	0	0	9	0	0	10

(b) DE_PLSA

Actual Classes

Predicted Classes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	9	2	1	2	2	0	1	0	0	0	0	0	0	0	0	1	0	0
2	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	9	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
4	1	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	6	0	2	1	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	9	0	0	0	1	0	0	3	2	0	0	0	1
7	0	1	0	1	0	0	6	0	2	0	0	3	0	0	0	0	0	0
8	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	1	0	7	0	0	0	0	0	0	0	0	0
10	0	0	0	0	1	0	0	0	0	6	0	0	2	1	0	0	0	0
11	0	0	0	0	1	0	0	0	0	1	8	0	1	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	1	2	2	0	0	0	1
14	0	0	0	0	0	0	0	0	0	2	0	0	0	3	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	1	0	0	10	0	0	2
16	0	1	0	0	0	1	0	0	0	0	1	1	2	2	0	9	2	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0
18	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	1	6

(c) DW_LDA

Actual Classes

Predicted Classes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	9	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
3	0	0	10	0	0	0	0	0	0	0	0	0	0	0	2	1	1	0
4	0	0	0	10	0	0	0	0	3	0	0	0	0	0	0	0	0	0
5	0	1	0	0	10	0	3	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	1	0
8	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
13	0	0	0	0	0	1	0	0	0	1	0	0	9	3	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	1	7	0	0	0	0
15	0	0	0	0	0	1	0	0	0	1	0	1	0	0	1	0	0	2
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0
18	0	0	0	0	0	0	0	0	0	0	7	0	0	7	0	0	0	8

(d) DE_LDA

but only some noise whose meaning is hard to be described; for tube class, although its typical audio event (the tube sound) occurs in most of the 10 audio documents, due to the reason that the tube sound is also the typical audio event of the tubestation class, it would easily be confused with the tubestation class.

From Table 3 it can be seen that on DEMAND dataset, DE_PLSA performs better than DW_PLSA in recognizing most of the audio scene classes, and DE_LDA performs better than DW_LDA in recognizing most of the audio scene classes as well. However, both DE_PLSA and DE_LDA perform badly on the audio scene classes of station and traffic. For the station class, its typical audio event is the sound of train, and for the metro class, its typical audio event is the sound of train as well, and then after the topic analysis, there would be much confusion between the topic distributions of these two classes; for the traffic class, its typical audio event is the sound of bus, and for the metro class, its typical audio event is the sound of train, intuitively, these two different kinds of audio event would be very likely to reflect the same topic—traveling, and then it would cause topic confusion between these two classes. Maybe it is due to the above reasons that many audio documents of station and traffic have been misclassified as metro, as is shown in sub-table (b) and (d) of Table 3.

The proposed algorithm relies on the document-event co-occurrence matrix to do topic analysis. Form the above analysis it can be seen that, for a specific audio scene class, if the typical audio events occur frequently in both the training and test set, then the proposed algorithm will perform very well, however, if the typical audio events occur only in a small part of the training and test set, such as the park class in the AASP dataset, or if there are almost no typical audio events, such as the restaurant and supermarket class in the AASP dataset, then the proposed algorithm will perform poorly. In this case, combining the document-word co-occurrence matrix with the document-event co-occurrence matrix to do topic analysis may be more proper.

Most of the current topic model based ASR algorithms have adopted SVM as the classification model [18–20]; for our proposed algorithm, in order to be fair in comparison with the algorithms proposed in [19] and [20], it also adopts SVM as the classification model. In this section, using the topic distribution extracted based on the document-event co-occurrence matrix as the feature set of

Table 4
The performance of SVM, Random Forest and AdaBoost on AASP and DEMAND dataset.

Dataset	Topic model	Model	Accuracy (%)	St. dev.
AASP	PLSA	SVM	61.0	6.5
		Random Forest	62.0	8.4
		AdaBoost	**63.0**	7.6
	LDA	SVM	54.0	2.2
		Random Forest	53.0	4.5
		AdaBoost	**55.0**	3.5
DEMAND	PLSA	SVM	83.3	3.9
		Random Forest	83.9	4.5
		AdaBoost	**84.5**	6.7
	LDA	SVM	78.9	9.7
		Random Forest	77.8	8.1
		AdaBoost	**80.6**	9.8

audio documents, beside SVM, we will use another two ensemble classifiers, i.e. Random Forest and AdaBoost, to do classification, and will make a comparison of these three classification models.

For Random Forest and AdaBoost, their performance is also evaluated through 5-fold cross validation; in each iteration, the method of splitting the dataset into training and test set is the same as that of using SVM as the classification model. For Random Forest, the number of trees in the ensemble and the number of variables to consider at each node are determined through the grid search method; the criterion used for splitting the nodes is information gain. For AdaBoost, K-means is used as the weak learner for the reason that it is the one of simplest learner which works for both discrete and continuous data. AdaBoost is a binary classification model, in order to do multiclass classification, one-vs-one strategy is used. The performance of these three classification models on both datasets is shown in Table 4.

From Table 4. it can be seen that among SVM, Random Forest and AdaBoost, AdaBoost performs best; Random Forest performs a little better than SVM in some cases, and a little worse in other cases, generally, Random Forest and SVM perform much equally. For AdaBoost, it integrates several weak classifiers, and then promotes them into a strong classifier, in this way it can reach a much lower generalization error rate; for Random Forest, it also has the advantage of ensemble, but the features used in each single classifier are part of the topic distribution selected randomly from the whole topic distribution; the topic distribution as a whole may describe the audio documents well, but part of it may not. In summary, when performing ASR based on topic model, ensemble classifier may be a good choice, but the features used in each single classifier should be the whole topic distribution, rather than part of it.

5.2.3. The necessity and the effectiveness of the proposed weighting method

For the proposed DE_PLSA/DE_LDA, the event distribution of each audio document is weighted with the coefficient designed based on the entropy of the events. In order to verify the necessity and the effectiveness of the proposed weighting method, in this section we will compare our proposed ASR algorithm with another three ASR algorithms, one is the ASR algorithm whose event distribution is not weighted, and the other two are the ASR algorithms whose event distribution is weighted by another two different weighting methods. We use DE_PLSA_NW/DE_LDA_NW to denote the ASR algorithm whose event distribution is not weighted; use DE_PLSA_UW/DE_LDA_UW to denote the ASR algorithm whose event distribution is weighted with uniform weight, specifically, if there are m different events in the event distribution, then the weight value of each event is $1/m$. We use DE_PLSA_AW/DE_LDA_AW to denote the ASR algorithm whose event distribution is weighted with the coefficient determined through AdaBoost feature selection; the AdaBoost feature selection method proposed in [26] is adopted to determine the weighting coefficient. In [26], the weak classifier selection mechanism of AdaBoost is used to select the feature components; we use the event distribution of the audio document as the input of this feature selection method; after feature selection, the weight values of the feature components are normalized to [0,1], and then are used as the weighting coefficient. For these three ASR algorithms, except for the weighting setting, their other settings are the same as that of DE_PLSA/DE_LDA. The classification performance of DE_PLSA, DE_PLSA_NW, DE_PLSA_UW and DE_PLSA_AW on AASP and DEMAND dataset and the classification performance of DE_LDA, DE_LDA_NW, DE_LDA_UW and DE_LDA_AW on AASP and DEMAND dataset are shown in Table 5. All classification performance is obtained through 5-fold cross validation.

Table 5
The classification performance of DE_PLSA, DE_PLSA_NW, DE_PLSA_UW and DE_PLSA_AW on AASP and DEMAND dataset, and the classification performance of DE_LDA, DE_LDA_NW, DE_LDA_UW and DE_LDA_AW on AASP and DEMAND dataset.

Dataset	Topic model	Algorithm	Accuracy (%)	St. dev.
AASP	PLSA	DE_PLSA	**61**	6.5
		DE_PLSA_NW	48	10.4
		DE_PLSA_UW	49	8.9
		DE_PLSA_AW	44	7.4
	LDA	DE_LDA	**54**	2.2
		DE_LDA_NW	47	6.7
		DE_LDA_UW	49	8.2
		DE_LDA_AW	49	10.8
DEMAND	PLSA	DE_PLSA	**83.3**	3.9
		DE_PLSA_NW	71.1	7.0
		DE_PLSA_UW	72.2	7.6
		DE_PLSA_AW	71.7	4.1
	LDA	DE_LDA	**78.9**	9.7
		DE_LDA_NW	70.0	5.3
		DE_LDA_UW	72.2	5.9
		DE_LDA_AW	62.8	3.2

From Table 5 it can be seen that on both datasets, DE_PLSA performs better than DE_PLSA_NW, and DE_LDA performs better than DE_LDA_NW, which illustrates that it is very necessary to weight the event distribution of the audio documents; Among DE_PLSA, DE_PLSA_UW and DE_PLSA_AW, it is DE_PLSA that performs best; among DE_LDA, DE_LDA_UW and DE_LDA_AW, it is DE_LDA that performs best, which illustrates that the proposed weighting method is very effective. In this work, we weight the event distribution based on the topic entropy of the events. The event with prominent topics is assigned with larger weight value, and the event that is common to many topics is assigned with smaller weight value. In this way, the proposed algorithm hopes to emphasize the events which are important in reflecting the unique topics of the audio documents, and suppress the events that are common to many topics. Experimental results in Table 5 have confirmed the effectiveness of such weighting. After weighting, the extracted topic distribution can indeed express the audio documents better. For DE_PLSA_UW and DE_LDA_UW, through the usage of uniform weighting, they treat each audio event equally, and this could not highlight the events that are important in reflecting the topics, and then their extracted topic distribution can not represent the audio documents well. For DE_PLSA_AW and DE_LDA_AW, they weight the event distribution with the coefficient determined through AdaBoost feature selection; such weighting coefficient can reflect the importance of the event in direct recognition, but it can not reflect the importance of the event in reflecting the topics of the audio documents, in this way, such weighting method may be good for the ASR algorithms which take the event distribution as

Table 6
The sensitivity values of the classifiers under the feature set of TD-DW and TD-DE on AASP and DEMAND dataset. The false-positive rate is set to a fixed 50%. When extracting the feature set of TD-DW and TD-DE, PLSA and LDA are used as the topic model respectively.

Dataset	Topic model	Feature set	Sensitivity (%)
AASP	PLSA	TD-DW	90
		TD-DE	**100**
	LDA	TD-DW	50
		TD-DE	**80**
DEMAND	PLSA	TD-DW	**100**
		TD-DE	**100**
	LDA	TD-DW	90
		TD-DE	**100**

the feature set for recognition, but it may not be good for the ASR algorithms which take the topic distribution as the feature set for recognition.

5.2.4. Feature sensitivity analysis

Most of the current topic model based ASR algorithms use the topic distribution extracted based on the document-word co-occurrence matrix as the feature set for recognition, while in this work, we use the topic distribution extracted based on the document-event co-occurrence matrix as the feature set for recognition. In this section we will perform feature sensitivity analysis to show its effectiveness. To do so, for each dataset, we use any two of the classes to train the binary SVM classifier, and to do binary classification through 5-fold cross validation. For each binary classifier, we tune the classification threshold to achieve a fixed false-positive rate of 50%, and then evaluate its sensitivity (also called true-positive rate). All combinations of two classes are used, and the average sensitivity value of all these binary classifiers is taken as the final result. The topic distribution extracted based on the document-word co-occurrence matrix is denoted as TD-DW, and the topic distribution extracted based on the document-event co-occurrence matrix is denoted as TD-DE. TD-DW and TD-DE are used to train the classifiers respectively. When extracting the topic distribution, PLSA and LDA are used as the topic model respectively. The sensitivity values of TD-DW and TD-DE under the topic model of PLSA and LDA are listed in Table 6.

Form Table 6 it can be seen that on both datasets, taking TD-DE as the feature set can achieve a sensitivity of no less than 80%, and it is larger than or equal to that taking TD-DW as the feature set, therefore, it validates that the proposed TD-DE is more effective than TD-DW. For TD-DE, the reason for its effectiveness maybe lies in two aspects: first, it is extracted based on the document-event co-occurrence matrix, and using the document-event co-occurrence matrix for topic analysis is more reasonable than using the document-word co-occurrence matrix; second, we have proposed a weighting method to emphasize the audio events that are important in reflecting the unique topics of the audio documents, and to suppress the audio events that are common to many topics, in this way, TD-DE can represent the audio documents better.

5.2.5. The rationality of obtaining both the document-event co-occurrence matrix of the training set and the document-event co-occurrence matrix of the test set through matrix factorization

For our proposed ASR algorithm, both the document-event co-occurrence matrix of the training set (that is B_{train}) and the document-event co-occurrence matrix of the test set (that is B_{test}) are obtained through PLSA matrix factorization; here we denote this scheme of obtaining B_{train} and B_{test} as scheme 1. In Section 4.2 we have discussed that actually there is another scheme to obtain B_{train} and B_{test}, that is, B_{train} is obtained through counting, and B_{test} is obtained through matrix factorization; we denote this scheme as scheme 2. In this section we will make a comparison of these two schemes.

Table 7
The classification performance of the systems using scheme 1 and scheme 2 respectively on AASP and DEMAND dataset.

Dataset	Topic model	Scheme	Accuracy (%)	St. dev.
AASP	PLSA	Scheme 1	**61.0**	6.5
		Scheme 2	59.0	7.4
	LDA	Scheme 1	**54.0**	2.2
		Scheme 2	51	4.2
DEMAND	PLSA	Scheme 1	**83.3**	3.9
		Scheme 2	80.6	3.4
	LDA	Scheme 1	**78.9**	9.7
		Scheme 2	74.5	5.3

For scheme 1, the experiments have been done in Section 5.2.2, i.e. the experiments which were done to test the performance of the proposed DE_PLSA/DE_LDA. For scheme 2, the experimental setting is the same as that of scheme 1. The classification performance of the systems using scheme 1 and scheme 2 respectively on both datasets is shown in Table 7. From Table 7 it can be seen that on both datasets, under the two different topic models of PLSA and LDA, scheme 1 always performs better than scheme 2, therefore, in this work, scheme 1 is adopted. The reason for the worse performance of scheme 2 may be that: for scheme 2, B_{train} and B_{test} are obtained through different ways: one through counting and one through matrix factorization, which would then causes inconsistency between B_{train} and B_{test}, and this inconsistency would then decrease the recognition performance of the system.

6. Conclusions and future work

In this work, we have proposed a new ASR algorithm. Different from the current studies which use document-word co-occurrence matrix for topic analysis, the proposed algorithm has adopted the document-event co-occurrence matrix for topic analysis. In order to obtain the document-event co-occurrence matrix in an easier way, and also in order to make the document-event co-occurrence matrix of the training set to be more consistent with that of the test set, in this work we propose a matrix-factorization method to obtain it. Realizing that the importance of the audio events in reflecting the topics of the audio documents is different, in order to emphasize the audio events that are important in reflecting the unique topics of the audio documents, and to suppress the audio events that are common to many topics, in this work we propose a weighting method, and use it to weight the event distribution of the audio documents.

The above innovative ideas are not limited to ASR. Taking video classification for example, in video documents there are also various events taking the form of image and audio, and then the algorithm proposed in this work can be easily extended to video classification. Likewise, the proposed algorithm can also be extended to many other classification fields, and the key point of the algorithm extension is how to define the “events” in different classification fields.

During experiments we found that on AASP dataset, the proposed algorithm performed worse than many algorithms which had been submitted to IEEE AASP challenge, this might be because that we had used different data processing methods and different parameter setting methods etc. In future work, we will do efforts to further improve the proposed algorithm, but in this work, at least we have verified that using the document-event co-occurrence matrix for topic analysis is better than using the document-word co-occurrence matrix.

Though the proposed algorithm can achieve better results, its limitations should be noticed:

(1) To count the event-word co-occurrence matrix, the audio documents in the training set should be manually annotated, and for the time intervals which contain multiple events, more than one label is annotated. In this way, when the dataset is large, the manual annotation workload will be large too. Besides, for the time intervals which contain multiple events, since more than one label is annotated, there inevitably exists the case that the annotation results of different annotators and different annotation sessions are different. When the dataset is large, to reduce the manual annotation workload, one solution is to select parts of the documents for annotation, and then use just these part annotation results for statistics. In order to reduce the inconsistency between different annotators and different annotation sessions, at least 3 annotators are required to annotate the same document, and the average annotation results are then used as the final annotation results.
(2) The proposed algorithm relies on the document-event co-occurrence matrix to do topic analysis. For a specific audio scene class, if the typical audio events occur only in the training or test set, or if there are almost no typical audio events in the training and test set, then the proposed algorithm will perform poorly. In this case, combining the document-word co-occurrence matrix with the document-event co-occurrence matrix to do topic analysis may be more proper. In future work, how to overcome the event inconsistency problem between the training and test set and how to overcome the problem of lacking typical events is an important research direction.

With the popularity of deep learning, currently there are some extensions of topic model which combine it with the neural network. For example, Wan et al. [27] proposed a hybrid neural network-latent topic model in which the neural network acts as a trainable feature extractor, and the topic model captures the group structure of the data; Cao et al. [28] proposed a novel neural topic model in which deep learning techniques are used to conduct inference under the topic model framework. In future work, we will take the above models as a reference, and try to use the neural network to extend our system; with the advantages of both neural networks and topic models, it surely can further improve the recognition performance.

Acknowledgments

This work has been jointly supported by the Project of National Natural Science Foundation of China (No. 61401259, No. 61362031, No. 61471226, No. 61401258, No. 61501283, No. 61305015), China Postdoctoral Science Foundation Funded Project (No. 2015M582128, No. 2015M582129, No. 2015M580591), Natural Science Foundation for Distinguished Young Scholars of Shandong Province (JQ201516), Project of Shandong Province Higher Educational Science and Technology Program (J16LN21), Research Fund for Young and Middle-aged Scientists of Shandong Province (ZR2016FB25), Shandong Province Key Research and Development Program (2016GGX101022), and Natural Science Foundation of Shandong Province (ZR2015PF012).

References

[1] Y. Leng, C. Sun, X. Xu, et al., Employing unlabeled data to improve the classification performance of SVM, and its application in audio event classification, Knowl. Based Syst. 98 (2016) 117–129.
[2] Y. Leng, C. Sun, C. Cheng, et al., Classification of overlapped audio events based on AT, PLSA, and the combination of them, Radioengineering 24 (2) (2015) 593–603.
[3] Y. Leng, G. Qi, X. Xu, et al., A BIC based initial training set selection algorithm for active learning and its application in audio detection, Radioengineering 22 (2) (2013) 638–649.
[4] W.H. Choi, S.I. Kim, M.S. Keum, et al., Acoustic and visual signal based context awareness system for mobile application, IEEE Trans. Consum. Electron. 57 (2) (2011) 738–746.
[5] D.V. Smith, M.S. Shahriar, A context aware sound classifier applied to prawn feed monitoring and energy disaggregation, Knowl. Based Syst. 52 (2013) 21–31.
[6] J.C. Wang, H.P. Lee, J.F. Wang, et al., Robust environmental sound recognition for home automation, IEEE Trans. Autom. Sci. Eng. 5 (1) (2008) 25–31.
[7] T. Hofmann, Probabilistic latent semantic analysis, in: Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence, 1999, pp. 289–296.
[8] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.
[9] H. Xu, F. Zhang, W. Wang, Implicit feature identification in Chinese reviews using explicit topic mining model, Knowl. Based Syst. 76 (2015) 166–175.
[10] H. Zhang, G. Zhong, Improving short text classification by learning vector representations of both words and hidden topics, Knowl. Based Syst. 102 (2016) 76–86.
[11] K. Pliakos, C. Kotropoulos, PLSA driven image annotation, classification, and tourism recommendation, in: Proceedings of IEEE International Conference on Image Processing (ICIP), 2014, pp. 3003–3007.
[12] G. Zhou, J. Zhao, T. He, et al., An empirical study of topic-sensitive probabilistic model for expert finding in question answer communities, Knowl. Based Syst. 66 (2014) 136–145.
[13] T.J. Hazen, Latent topic modeling for audio corpus summarization, in: Proceedings of Conference of the International Speech Communication Association (INTERSPEECH), 2011, pp. 913–916.
[14] A. Mesaros, T. Heittola, A. Klapuri, Latent semantic analysis in sound event detection, in: Proceedings of 19th European Signal Processing Conference (EUSIPCO), 2011, pp. 1307–1311.
[15] P. Hu, W. Liu, W. Jiang, et al., Latent topic model for audio retrieval, Pattern Recognit. 47 (3) (2014) 1138–1143.
[16] S. Kim, S. Narayanan, S. Sundaram, Acoustic topic model for audio information retrieval, in: Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2009, pp. 37–40.
[17] S. Kim, P. Georgiou, S. Narayanan, Latent acoustic topic models for unstructured audio classification, APSIPA Trans. Signal Inf. Process. 1 (1) (2012) e6.
[18] Y. Peng, Z. Lu, J. Xiao, Semantic concept annotation based on audio PLSA model, in: Proceedings of the 17th ACM International Conference on Multimedia, 2009, pp. 841–844.
[19] K. Lee, D.P.W. Ellis, Audio-based semantic concept classification for consumer video, IEEE Trans. Audio Speech Lang. Process. 18 (6) (2010) 1406–1416.
[20] S. Kim, S. Sundaram, P. Georgiou, et al., Audio scene understanding using topic models, in: Proceedings of the Neural Information Processing Systems (NIPS) Workshop, 2009.
[21] M. Steyvers, T. Griffiths, Probabilistic topic models, in: Handbook of Latent Semantic Analysis, 427, 2007, pp. 424–440.
[22] D.P. Ellis, K. Lee, Minimal-impact audio-based personal archives, in: Proceedings of the 1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences, 2004, pp. 39–47.
[23] Y. Leng, X. Xu, G. Qi, Combining active learning and semi-supervised learning to construct SVM classifier, Knowl. Based Syst. 44 (2013) 121–131.
[24] D. Stowell, D. Giannoulis, E. Benetos, et al., Detection and classification of acoustic scenes and events, IEEE Trans. Multimedia 17 (10) (2015) 1733–1746.
[25] J. Thiemann, N. Ito, E. Vincent, The diverse environments multi-channel acoustic noise database: a database of multichannel environmental noise recordings, J. Acoust. Soc. Am. 133 (5) (2013) 3591-3591.
[26] X. Zhuang, X. Zhou, M.A. Hasegawa-Johnson, et al., Real-world acoustic event detection, Pattern Recognit. Lett. 31 (12) (2010) 1543–1551.
[27] L. Wan, L. Zhu, R. Fergus, A hybrid neural network-latent topic model, in: Proceedings of the 15th International Conference on Artificial Intelligence and Statistics, 2012, pp. 1287–1294.
[28] Z. Cao, S. Li, Y. Liu, et al., A novel neural topic model and its supervised extension, in: Proceedings of 29th AAAI Conference on Artificial Intelligence, 2015, pp. 2210–2216.