Bayesian Learning for Latent Semantic Analysis

Jen-Tzung Chien, Meng-Sung Wu and Chia-Sheng Wu

Department of Computer Science and Information Engineering
National Cheng Kung University, Tainan, Taiwan 70101, ROC
{chien,jackywu,cswu}@chien.csie.ncku.edu.tw

10.21437/Interspeech.2005-9

Abstract

Probabilistic latent semantic analysis (PLSA) is a popular approach to text modeling where the semantics and statistics in documents can be effectively captured. In this paper, a novel Bayesian PLSA framework is presented. We focus on exploiting the incremental learning algorithm for solving the updating problem of new domain articles. This algorithm is developed to improve text modeling by incrementally extracting the up-to-date latent semantic information to match the changing domains at run time. The expectation-maximization (EM) algorithm is applied to resolve the quasi-Bayes (QB) estimate of PLSA parameters. The online PLSA is constructed to accomplish parameter estimation as well as hyperparameter updating. Compared to standard PLSA using maximum likelihood estimate, the proposed QB approach is capable of performing dynamic document indexing and classification. Also, we present the maximum *a posteriori* PLSA for corrective training. Experiments on evaluating model perplexities and classification accuracies demonstrate the superiority of using Bayesian PLSA.

1. Introduction

Recently, latent semantic analysis (LSA) [6] has been effectively applied for *n*-gram language modeling [1][5]. The basic idea of LSA aims to represent the words or documents in a low dimensional vector space consisting of the common semantic factors. Using LSA, all words and documents are mapped to the common semantic space, which is constructed via the singular value decomposition (SVD) of a word-by-document matrix. More attractively, the probabilistic LSA (PLSA) is presented to model the aspects in documents in probabilistic way that the document-word joint distributions are expressed conditionally on latent mixtures or topics [10][11]. PLSA text modeling was shown to achieve low perplexity. A topic-based language modeling based on PLSA was developed [9]. Furthermore, a latent Dirichlet allocation [3] method was exploited to deal with the issue of PLSA that the aspect models were estimated only for those documents appearing in training set. In this study, we strive for developing Bayesian PLSA aspect model for *adaptive text modeling and classification*. Our goal aims to establish the incremental learning and corrective training capabilities for PLSA. The adaptive PLSA modeling is fulfilled to recognize unknown documents with changing domains/topics.

The underlying concept of online PLSA is motivated from the principle of quasi-Bayes (QB) estimate, which has been successfully applied for speech recognition [4][12]. Instead of adaptive hidden Markov modeling (HMM) for speech recognition, we propose QB estimate of PLSA model where the model parameters are estimated by maximizing an approximate posterior distribution, or equivalently a product of likelihood function of currently observed documents and *a priori* density given the updating hyperparameters. The advantage of incremental learning highlights on continuously updating model parameters without waiting long history of batch documents. Computation and memory requirements can be substantially reduced. After simplification, the maximum *a posteriori* (MAP) PLSA is realized for batch model adaptation. Expectation-maximization (EM) algorithm [8] is adopted to resolve missing data or latent variable problem in Bayesian parameter estimation. In the experiments, we conduct evaluation of perplexity and document classification using Bayesian PLSA. The updating performance can be consistently improved when increasing number of adaptation documents.

2. Probabilistic Latent Semantic Analysis

PLSA is a general machine learning technique, which adopts the aspect model to represent the co-occurrence data associated with a topic or hidden variable $z_k \in Z = \{z_1, \ldots, z_K\}$. Let the text corpus Y consist of document-word pairs (d_i, w_j) collected from N documents $d_i \in \{d_1, \ldots, d_N\}$ with a vocabulary of M words $w_j \in \{w_1, \ldots, w_M\}$. The joint probability of observed pair (d_i, w_j) can be generated in asymmetric parameterization form [11]

$$P(d_i, w_j) = P(d_i) \sum_{k=1}^{K} P(w_j | z_k) P(z_k | d_i) , \qquad (1)$$

assuming that d_i and w_j are independent conditionally on the mixture of associated topic z_k. We can accumulate the log likelihood of overall training data $Y = \{d_i, w_j\}$ as follows

$$\log P(Y|\theta) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) , \qquad (2)$$

where $n(d_i, w_j)$ is the count of word w_j occurring in document d_i and θ is the PLSA parameter set $\theta = \{P(w_j|z_k), P(z_k|d_i)\}$. According to maximum likelihood (ML) estimate, PLSA parameters are obtained via maximizing accumulated log likelihood

$$\theta_{\mathrm{ML}} = \arg\max_{\theta} \log P(Y|\theta) . \qquad (3)$$

Due to the latent variable z_k appearing in PLSA, we should apply EM algorithm to solve ML parameter estimation. In E-step, we calculate an expectation function of new estimate $\hat{\theta} = \{\hat{P}(w_j|z_k), \hat{P}(z_k|d_i)\}$ over the latent variable z_k to find

$$Q(\hat{\theta}|\theta) = E_Z[\log P(Y, Z|\hat{\theta})|Y, \theta] =$$

$$\sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j) \sum_{k=1}^{K} P(z_k|d_i, w_j) \log[\hat{P}(w_j|z_k)\hat{P}(z_k|d_i)] . \quad (4)$$

The posterior probability using current estimate $\theta = \{P(w_j|z_k), P(z_k|d_i)\}$ is expressed by

$$P(z_k|d_i, w_j) = \frac{P(w_j|z_k)P(z_k|d_i)}{\sum_{l=1}^{K} P(w_j|z_l)P(z_l|d_i)} . \quad (5)$$

In M-step, we maximize $Q(\hat{\theta}|\theta)$ with respect to $\hat{\theta}$ and find ML new estimate $\hat{\theta}_{\mathrm{ML}}$ given by [11]

$$\hat{P}_{\mathrm{ML}}(w_j|z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j) P(z_k|d_i, w_j)}{\sum_{m=1}^{M}\sum_{i=1}^{N} n(d_i, w_m) P(z_k|d_i, w_m)} , \quad (6)$$

$$\hat{P}_{\mathrm{ML}}(z_k|d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j) P(z_k|d_i, w_j)}{\sum_{l=1}^{K}\sum_{j=1}^{M} n(d_i, w_j) P(z_l|d_i, w_j)} . \quad (7)$$

3. Bayesian Learning for PLSA Modeling

Practically, either SVD in LSA or ML parameters of (6)(7) in PLSA should be adaptive to deal with out-of-vocabulary or out-of-domain problem in an information system. We need to trace domain knowledge from new data collection so as to update LSA or PLSA model. Accordingly, the system robustness can be enhanced to handle changing document collections. Using LSA, the updating problems can be resolved via folding-in, SVD recomputing or SVD updating [2]. In what follows, we present the solutions to updating issues of PLSA model via Bayesian learning of model parameters.

3.1. PLSA Updating

Although updating problems have been investigated for SVD-based LSA framework, similar approaches can not applied for ML-based PLSA framework. To merge up-to-date knowledge into an existing PLSA model, we highlight on building adaptive PLSA for document modeling and classification. Our goal is to use the newly collected documents, called adaptation data X, to adapt an existing PLSA model to fit the domains of new documents or queries. Importantly, we apply Bayesian theory to develop two new adaptation paradigms for PLSA (1) *corrective training* and (2) *incremental learning*. In [4][12], Bayesian learning has been explored to adjust the existing speech HMM's to a new speaker for adaptive speech recognition. Herein, we present the maximum *a posteriori* (MAP) PLSA and quasi-Bayes (QB) PLSA to achieve corrective training and incremental learning for adaptive text mining, respectively.

3.2. MAP Estimation for Corrective Training

According to MAP estimation, PLSA parameters θ are estimated by maximizing *a posteriori* probability $P(\theta|X)$, or correspondingly the sum of logarithms of a likelihood function $P(X|\theta)$ and a prior density $g(\theta)$

$$\theta_{\mathrm{MAP}} = \arg\max_{\theta} P(\theta|X) = \arg\max_{\theta} \log P(X|\theta) + \log g(\theta) . \quad (8)$$

Prior density represents the randomness of probability parameters $\theta = \{P(w_j|z_k), P(z_k|d_i)\}$. Basically, the definition of prior distribution plays a crucial role in Bayesian learning. As suggested in [7], the choice of *conjugate prior* is good for Bayesian inference. We will show you later two attractive properties of using conjugate prior: 1) a closed-form solution for rapid learning and 2) a reproducible prior/posterior pair for incremental learning. As we know, Dirichlet density is referred as the conjugate prior for probability parameters or multinomial observations [7]. Assuming the variables $P(w_j|z_k)$ and $P(z_k|d_i)$ are independent, the prior density of overall parameters is expressed by

$$g(\theta) \propto \prod_{k=1}^{K}\left[\prod_{j=1}^{M} P(w_j|z_k)^{\alpha_{j,k}-1} \prod_{i=1}^{N} P(z_k|d_i)^{\beta_{k,i}-1}\right], \quad (9)$$

where $\varphi = \{\alpha_{j,k}, \beta_{k,i}\}$ are hyperparameters of Dirichlet densities. Again, we apply EM algorithm to iteratively calculate the posterior expectation function $R(\hat{\theta}|\theta)$ (E-step) and maximize it with respect to $\hat{\theta}$ so as to find new MAP estimates $\hat{\theta}_{\mathrm{MAP}}$ (M-step). Because the optimization is performed subject to the constraints $\sum_{j=1}^{M}\hat{P}(w_j|z_k) = 1$ and $\sum_{k=1}^{K}\hat{P}(z_k|d_i) = 1$, we obey the Lagrange optimization procedure and form the modified expectation function

$$\begin{aligned}\tilde{R}(\hat{\theta}|\theta) \propto & \sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j) \sum_{k=1}^{K} P(z_k|d_i, w_j) \log[\hat{P}(w_j|z_k)\hat{P}(z_k|d_i)] \\ & + \sum_{j=1}^{M}\sum_{k=1}^{K} (\alpha_{j,k} - 1) \log \hat{P}(w_j|z_k) + \sum_{i=1}^{N}\sum_{k=1}^{K} (\beta_{k,i} - 1) \log \hat{P}(z_k|d_i) \\ & + \eta_w (1 - \sum_{j=1}^{M} \hat{P}(w_j|z_k)) + \eta_d (1 - \sum_{k=1}^{K} \hat{P}(z_k|d_i))\end{aligned} \quad (10)$$

where η_d and η_w are two Lagrange multipliers. Then, we differentiate $\tilde{R}(\hat{\theta}|\theta)$ with respect to $\hat{P}(w_j|z_k)$ and $\hat{P}(z_k|d_i)$ to obtain new MAP estimates in closed-form solution

$$\hat{P}_{\mathrm{MAP}}(w_j|z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j) P(z_k|d_i, w_j) + (\alpha_{j,k} - 1)}{\sum_{m=1}^{M}[\sum_{i=1}^{N} n(d_i, w_m) P(z_k|d_i, w_m) + (\alpha_{j,m} - 1)]} \quad (11)$$

$$\hat{P}_{\mathrm{MAP}}(z_k|d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j) P(z_k|d_i, w_j) + (\beta_{k,i} - 1)}{n(d_i) + \sum_{l=1}^{K} (\beta_{l,i} - 1)} . \quad (12)$$

This MAP PLSA algorithm is designed for corrective training or batch adaptation, which adapts the existing PLSA parameters θ to θ_{MAP} using newly collected documents X in batch mode. Apparently, new parameter θ_{MAP} could perform better than θ when classifying future documents with new topics and terms. Due to the closed-form solution, we don't need to use descent algorithm to find optimal parameters. Rapid adaptation can be achieved.

3.3. QB Estimation for Incremental Learning

However, in an online information system, we need to continuously update system parameters with new words and topics. Such system can be robust for pattern recognition at different epochs. Because the system is updated at each epoch, the out-of-date words or documents will be gradually faded away or down dated from system parameters. In MAP PLSA,

there is no updating mechanism designed for online PLSA adaptation. Hereafter, we present the quasi-Bayes (QB) estimation to fulfill PLSA incremental learning. In general, at the *n*th epoch, we maximize the posterior probability using the sequence of documents $\chi^n = \{X_1, \cdots, X_n\}$ through [4][12]

$$\theta_{\mathrm{QB}}^{(n)} = \arg\max_{\theta} P(\theta|\chi^n) = \arg\max_{\theta} P(X_n|\theta) P(\theta|\chi^{n-1}) \cong \arg\max_{\theta} P(X_n|\theta) g(\theta|\varphi^{(n-1)}) \quad (13)$$

where the posterior density $P(\theta|\chi^{n-1})$ is approximated by the closest tractable prior density $g(\theta|\varphi^{(n-1)})$ with hyperparameters $\varphi^{(n-1)}$ evolved from historical documents χ^{n-1}. Attractively, QB estimation provides a recursive learning mechanism of PLSA parameters $\theta^{(1)}, \cdots, \theta^{(n)}$ from incrementally observed block documents $X_1, \cdots, X_n$. At each epoch, we use the current block of documents X_n and the accumulated statistics $\varphi^{(n-1)}$ to update PLSA parameters to $\theta_{\mathrm{QB}}^{(n)}$. After updating hyperparameters $\varphi^{(n-1)} \to \varphi^{(n)}$, the current observation documents $X_n = \{d_i^{(n)}, w_j^{(n)}\}$ are discarded without storage. As compared to MAP PLSA, the key difference using QB PLSA is due to the *updating of hyperparameters*. If we substitute the hyperparameters $\varphi^{(n-1)} = \{\alpha_{j,k}^{(n-1)}, \beta_{k,i}^{(n-1)}\}$ into (11)(12), the corresponding QB estimate $\theta_{\mathrm{QB}}^{(n)} = \{P_{\mathrm{QB}}(w_j^{(n)}|z_k), P_{\mathrm{QB}}(z_k|d_i^{(n)})\}$ can be obtained. Because EM algorithm is employed, the updating of hyperparameters can be derived in E-step of QB estimation. By introducing the latent variables $Z = \{z_k\}$, the expectation of the logarithm of posterior distribution $R(\hat{\theta}^{(n)}|\theta^{(n)})$ is expanded. After careful arrangement, the exponential of posterior expectation function can be expressed as a new Dirichlet distribution

$$\exp\{R(\hat{\theta}^{(n)}|\theta^{(n)})\} \propto \prod_{k=1}^{K}\left[\prod_{j=1}^{M} \hat{P}^{(n)}(w_j^{(n)}|z_k)^{\alpha_{j,k}^{(n)}-1} \prod_{i=1}^{N} \hat{P}^{(n)}(z_k|d_i^{(n)})^{\beta_{k,i}^{(n)}-1}\right], \quad (14)$$

with new hyperparameters $\varphi^{(n)} = \{\alpha_{j,k}^{(n)}, \beta_{k,i}^{(n)}\}$ given by

$$\alpha_{j,k}^{(n)} = \sum_{i=1}^{N} n(d_i^{(n)}, w_j^{(n)}) P^{(n)}(z_k|d_i^{(n)}, w_j^{(n)}) + \alpha_{j,k}^{(n-1)}, \quad (15)$$

$$\beta_{k,i}^{(n)} = \sum_{j=1}^{M} n(d_i^{(n)}, w_j^{(n)}) P^{(n)}(z_k|d_i^{(n)}, w_j^{(n)}) + \beta_{k,i}^{(n-1)}. \quad (16)$$

Interestingly, a *reproducible prior/posterior pair* is generated to build the updating mechanism of hyperparameters. In (15)(16), the new hyperparameters $\alpha_{j,k}^{(n)}$ and $\beta_{k,i}^{(n)}$ are obtained by interpolating the previous hyperparameters $\alpha_{j,k}^{(n-1)}$ and $\beta_{k,i}^{(n-1)}$ with the accumulated statistics of latent variable z_k related to word $w_j^{(n)}$ and document $d_i^{(n)}$, respectively.

3.4. Estimation of Initial Hyperparameters

Either MAP PLSA or QB PLSA, it is critical to estimate initial hyperparameters $\varphi^{(0)} = \{\alpha_{j,k}^{(0)}, \beta_{k,i}^{(0)}\}$. Typically, the hyperparameters can be estimated from training data in an empirical Bayes sense. Nevertheless, the estimation of initial hyperparameters is an open issue in Bayesian learning. Without loss of generality, herein, we adopt the estimation of initial hyperparameters for Dirichlet density introduced by Huo and Lee [12]

$$\alpha_{j,k}^{(0)} = 1 + \sum_{i=1}^{N} P(z_k|d_i, w_j), \quad (17)$$

$$\beta_{k,i}^{(0)} = 1 + \sum_{j=1}^{M} P(z_k|d_i, w_j). \quad (18)$$

4. Experiments

In the experiments, we evaluate the performance of proposed methods using two public-domain document collections. One was MED corpus [10] and the other was Reuters-21578 [3]. Before PLSA modeling, we performed preprocessing stages of stemming and stop word removal for all documents.

4.1. Evaluation of Model Perplexity

When evaluating the perplexity performance, we used MED corpus containing 1,033 medical abstracts with 30 queries and 7,014 unique terms collected from National Library of Medicine. Without loss of generalization, we adopted 433 abstracts for ML PLSA model training and the remaining 600 abstracts for MAP corrective training or QB incremental learning. A query subset served as the test data for evaluation of model perplexity. To examine the effect on different numbers of adaptation data (N), we calculated perplexities for cases of N=200, 400 and 600. In case of N=600, MAP PLSA performed one learning epoch using all adaptation data while QB PLSA performed three learning epochs with 200 adaptation data at each epoch. Number of latent variables was fixed to be $K = 8$. Without model adaptation, the baseline perplexity is 146.7. As shown in Figure 1, the performance of MAP PLSA and QB PLSA are improved greatly. Perplexities are consistently reduced when increasing number of adaptation data. QB PLSA achieves lower perplexities compared to MAP PLSA. For the case of N=600, QB PLSA attained perplexity of 53.9 which is better than 63 using MAP PLSA. Also, in Figure 2, we compare computation times (measured in seconds) of MAP PLSA and QB PLSA for different numbers of adaptation data. Computation times were measured using a personal computer with Pentium IV 2.4GHz CPU and 1 GB RAM. We can see that MAP PLSA increases computation cost when increasing adaptation data. However, QB PLSA computes parameters only using adaptation data at current learning epoch. QB PLSA is efficient at each epoch.

4.2. Experimental Results on Text Classification

We followed the ModApte split of Reuters-21578 data set in which 7,195 documents were used for training or adaptation and 2,790 documents were used for classification. We further partitioned 7,195 documents into 4,270 documents for training and the remaining 2,925 documents for QB incremental adaptation. The corpus contained 13,353 unique words. Text classification was performed on ten most

populous categories. As shown in Figure 3, we report classification accuracies of QB PLSA for the cases of N=975, 1950 and 2925. N=0 implies baseline system. Numbers of latent variables $K = 64$ and $K = 128$ are investigated. We find that QB adaptation consistently improves performance when increasing number of adaptation data. In this case, $K = 64$ is sufficient for adaptive modeling and performs better than $K = 128$.

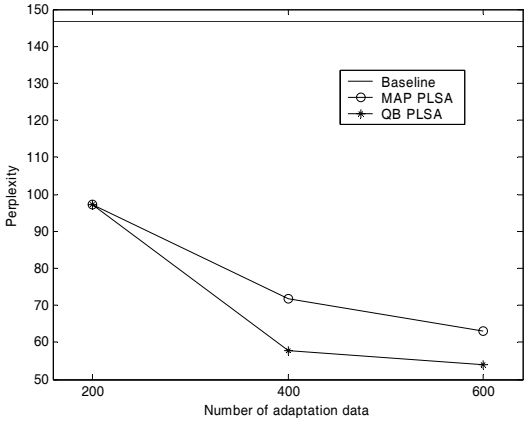

Figure 1: Perplexities for different numbers of adaptation data

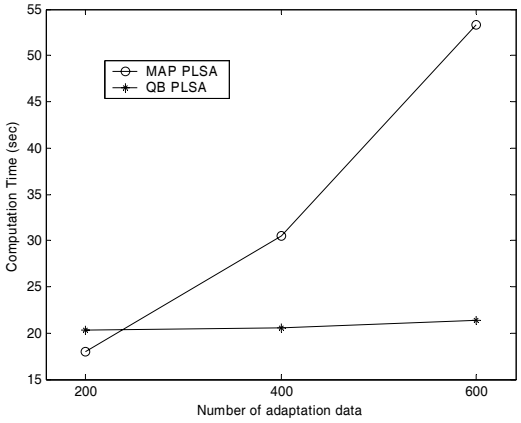

Figure 2: Computation times (sec) using MAP and QB PLSA

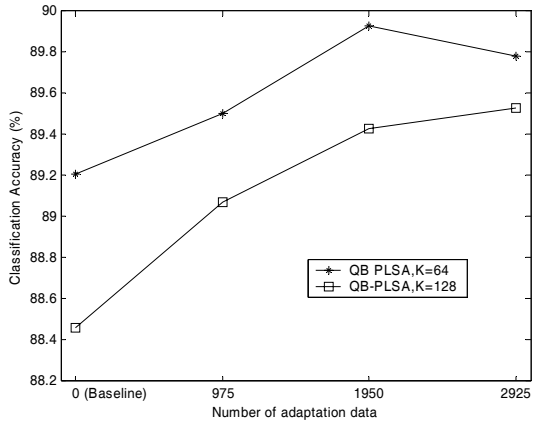

Figure 3: Classification accuracy (%) for different numbers of adaptation data and latent variables.

5. Conclusions

This paper presented an adaptive text modeling and classification approach for PLSA based information system. We performed batch adaptation and incremental adaptation through MAP PLSA and QB PLSA algorithms, respectively, without the needs of recomputing and storing whole adaptation data. We highlighted the contributions of incremental learning where new domain knowledge could be continuously updated and simultaneously the out-of-date words or documents would be faded away. In this manner, we did not only solve the updating problem but also the *downdating* problem. The experiments on evaluating model perplexity and classification accuracy illustrated the performance of using MAP PLSA and QB PLSA. In the future, we will develop Bayesian learning for other text modeling approaches. Extension of PLSA for bigram or trigram will be explored. Further work is also required to apply for spoken document classification and retrieval.

6. References

[1] J. R. Bellegarda, "Exploiting latent semantic information in statistical language modeling," *Proceeding of the IEEE*, vol. 88, No. 8, pp. 1279-1296, 2000.

[2] M. W. Berry, S. T. Dumais and G. W. O'Brien, "Using linear algebra for intelligent information retrieval", *SIAM Review*, vol. 37, no. 4, pp. 573-595, 1995.

[3] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent Dirichlet allocation", *Journal of Machine Learning Research*, vol. 3, no. 5, pp. 993-1022, 2003.

[4] J.-T. Chien, "Online hierarchical transformation of hidden Markov models for speech recognition", *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 6, pp. 656-667, 1999.

[5] J.-T. Chien, M.-S. Wu and H.-J. Peng, "On latent semantic language modeling and smoothing", *Proceedings of International Conference on Spoken Language Processing*, vol. 2, pp. 1373-1376, 2004.

[6] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer and R. Harshman, "Indexing by latent semantic analysis", *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391-407, 1990.

[7] M. H. DeGroot, *Optimal Statistical Decisions*, McGraw-Hill, 1970.

[8] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1-38, 1977.

[9] D. Gildea and T. Hofmann, "Topic based language models using EM", *Proceedings of 6th European Conference on Speech Communication and Technology*, pp. 2167-2170, 1999.

[10] T. Hofmann, "Probabilistic latent semantic indexing", *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 50-57, 1999.

[11] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis", *Machine Learning*, vol. 42, no. 1, pp. 177–196, 2001.

[12] Q. Huo and C.-H. Lee, "On-line adaptive learning of the continuous density hidden Markov model based on approximate recursive Bayes estimate", *IEEE Transactions on Speech and Audio Processing*, vol. 5, pp. 161-172, 1997.