Spoken Language Recognition in the Latent Topic Simplex

Kong Aik Lee, Chang Huai You, Ville Hautamäki, Anthony Larcher, and Haizhou Li

Human Language Technology Department, Institute for Infocomm Research,
Agency for Science, Technology and Research (A*STAR), Singapore
kalee@i2r.a-star.edu.sg

Abstract

This paper proposes the use of latent topic modeling for spoken language recognition, where a topic is defined as a discrete distribution over phone *n*-grams. The latent topics are trained in an unsupervised manner using the latent Dirichlet allocation (LDA) technique. Language recognition is then performed in a low dimensional simplex defined by the latent topics. We apply the Bhattacharyya measure to compute the *n*-gram similarity in the topic simplex. Our study shows that some of the latent topics are language specific while others exhibit multilingual characteristic. Experiment conducted on the NIST 2007 language detection task shows that language cues can be sufficiently preserved in the topic simplex.

Index Terms: phonotactic, language recognition, latent Dirichlet allocation

1. Introduction

In a spoken language recognition task, given a short segment of speech, the goal is to recognize the language identity corresponds to the speech segment. Research in this area was traditionally motivated to provide automatic solution to route an incoming call to an operator in the call center who is fluent in the corresponding language [1]. Excessive delay may lead to devastating consequence in emergency situation. More recently, spoken language recognition is found useful in spoken document retrieval [2] to automatically label overwhelming amount of spoken documents in the Web.

State-of-the-art spoken language recognition systems use either phonotactic or acoustic cues [3, 4, 5, 6]. In this regard, phone *n*-gram and *shifted-delta-cepstral* (SDC) are the most widely used. The central idea of using phonotactic cues is based on the assumption that each language possesses some unique phone patterns in terms of the order and frequency of occurrence of phones. These cues can be modeled using the so-called *n*-gram *language model* (LM) [3, 4, 5]. Another discriminative alternative is by presenting examples of phone *n*-gram probability to *support vector machine* (SVM) in which a hyperplane separating the target and non-target languages can be learned [6]. This paper is concerned with the use of phone *n*-gram probability with SVM for spoken language recognition.

One subtle problem in using the *n*-gram probability is the high dimensionality involved. For example, a typical phone recognizer with 40 phonetic units will lead to $(40)^3 = 64{,}000$ unique trigrams. This results in a very sparse estimate of *n*-gram probability with a lot of unseen *n*-grams, especially for the case of short segments. Reducing the number of unique *n*-grams, so as to reduce the number of parameters to be estimated, does not usually help as this reduces the resolution of the *n*-gram distribution. This problem has been traditionally tackled with smoothing or discounting methods [7], for examples, add-one smoothing, Witten-Bell discounting or Bayesian smoothing with Dirichlet prior [8]. More recently, latent topic model [9, 10] which has its roots in information retrieval for modeling text documents, has shown to be a promising solution. The central idea of a topic model is to confine the variability to a low dimensionality, very similar to the subspace methods [11, 12] used in speech processing.

This paper advocates the use of *latent Dirichlet allocation* (LDA) [10], the most widely used topic model, for modeling *n*-gram sequences. In this regard, we assume that the *n*-gram sequences are generated by a topic model characterized by a set of latent topics, each being a discrete distribution over the *n*-grams. Using this model, our aim in this paper is twofold. Firstly, by fitting the topic model onto a sufficiently large corpus, we aim to learn the hidden structure underlying the corpus, which corresponds to the hidden phonotactic constraints pertaining to individual languages and those common between languages. Secondly, we construct a low dimensional simplex using the latent topics, in which spoken language recognition can be done effective, if not better than in the original dimensionality. A simplex can be visualized as a lifted and bounded hyperplane. For language recognition to be done effectively in the simplex, we introduce an SVM kernel metric based on the Bhattacharyya measure. In the following sections, we demonstrate the LDA technique for modeling *n*-gram sequences in more detail in this first attempt of using topic model for language recognition.

2. Phone *n*-gram statistics as language cues

State-of-the-art phonotactic system comprises a phone recognition front-end and either a language model (LM) [3] or support vector machine (SVM) [6] back-end. The front-end uses a phone recognizer[1] to convert speech waveform $\mathcal{X}$ into phone sequence:

$$\mathcal{Y} = \arg\max_{Y} P(Y \mid \mathcal{X}, \mathcal{M}) \ , \qquad (1)$$

where $P(Y \mid \mathcal{X}, \mathcal{M})$ denotes the posterior probability of generating the phone sequence Y given the input $\mathcal{X}$ and the parameters of the phone recognizer, $\mathcal{M}$. Using the best phone sequence output $\mathcal{Y}$ from the recognizer, we then count the occurrences of *n*-grams: sub-sequences of *n* phone symbols. Take trigram (i.e., $n = 3$) for example, we count the number of times the symbol w_{t-2} is followed by w_{t-1} and w_t, which gives $C(w_{t-2}, w_{t-1}, w_t)$. Here, w_t represent any phone in the phone set. Let $C(w_{t-n+1}, \ldots, w_{t-1}, w_t)$ be the *n*-gram counts, the maximum likelihood (ML) estimate of the *n*-gram probability [7] is computed as

$$P(w_{t-n+1}, \ldots, w_{t-1}, w_t) = \frac{C(w_{t-n+1}, \ldots, w_{t-1}, w_t)}{N} \ , \qquad (2)$$

where N is the total number of *n*-grams observed in the phone sequence. Here, we treat individual *n*-gram as if it is a single event, which essentially means that the *n*-gram

[1] One can also use multiple phone recognizers in parallel (PPR), where the final decision is obtained by fusing the scores from parallel systems.

probability in (2) is a joint probability of *n* sub-events. This is slightly different from the LM approach, where the *n*-gram probability is modeled as the conditional probability, $P(w_t \mid w_{t-n+1}, \ldots, w_{t-1})$, in which the probability is conditioned on the preceding $(n-1)$ symbols $(w_{t-n+1}, \ldots, w_{t-1})$. In this paper, the *n*-gram probabilities serve as inputs to SVM, for which earlier study [6] has shown that joint probability *n*-gram model is more suitable.

The joint probability in (2) can be treated just like a unigram probability by letting $\tilde{w}_t = (w_{t-n+1}, \ldots, w_{t-1}, w_t)$. The symbol $\tilde{w}$ now represents any of the $V = M^n$ possible unique *n*-grams, where M is the number of unique phones. This is desirable as the latent Dirichlet allocation (LDA) originally proposed in [10] works on unigram probability over words from text documents. Though it is possible to use LDA on conditional probability [13], we devote the paper to the first option in this preliminary study.

3. Latent topics for language recognition

Latent Dirichlet allocation (LDA) was proposed in [10] originally for modeling the word occurrence frequency in text documents. Since text documents are essentially sequences of words, we use LDA to model *n*-gram sequences by treating the *n*-gram symbols as words in text documents.

3.1. Latent topics and topic simplex

We use similar notation $\tilde{w}$ to indicate an *n*-gram symbol as in previous section, and there are V number of those unique symbols. LDA assumes that an *n*-gram sequence $\mathcal{W} = (\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_t, \ldots, \tilde{w}_N)$ consisting of N observations is described by the following model:

$$P(\mathcal{W} \mid \boldsymbol{\alpha}, \mathbf{P}) = \int \mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \prod_{t=1}^{N} \left(\sum_{k=1}^{K} P(\tilde{w}_t \mid \theta_k) \theta_k \right) d\boldsymbol{\theta} . \quad (3)$$

In the model, the distributions $P(\tilde{w} \mid \theta_k)$, for k = 1, 2, …, K, are V-dimensional discrete distributions over all the unique *n*-grams. We refer to these distributions as the *latent topics*. The latent topics are linearly combined to give

$$P(\tilde{w} \mid \boldsymbol{\theta}) = \sum_{k=1}^{K} P(\tilde{w} \mid \theta_k) \theta_k , \quad (4)$$

from which the *n*-gram sequence $\mathcal{W}$ was generated. The weights, $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_K\}$, that determine the proportions of topics in the mixture are called the *latent factors*, and are assumed to follow a Dirichlet distribution:

$$\mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) = C(\boldsymbol{\alpha}) \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} . \quad (5)$$

Here, $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_K\}$ are the set of positive parameters and $C(\boldsymbol{\alpha})$ is the normalization factor to ensure that (5) is a legitimate density function. In the left hand side of (3), we use the $V \times K$ matrix $\mathbf{P}$ to represent the latent topics in a column-wise manner, where the (v,k)th element of the matrix is given by

$$\mathbf{P}(v,k) \equiv P(\tilde{w}_v \mid \theta_k) . \quad (6)$$

By using the latent factor model as described above, our aim is to discover the hidden phonotactic strands or cues underlying a particular language, and between languages, in terms of latent topics. We could also represent an *n*-gram sequence as the linear combination of latent topics as in (4). Since the number of topics, K, is usually much smaller than the number of unique symbols, V, the latent factors $\boldsymbol{\theta}$ can be used as a low-dimensional representation to the V-dimensional

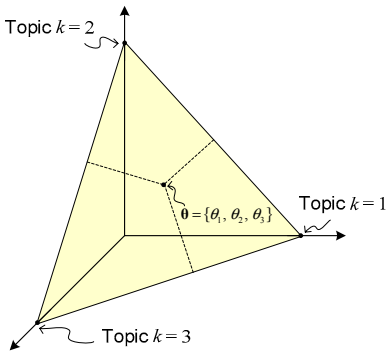

Figure 1: *A 2-dimensional simplex. At the vertices are the three latent topics. An n-gram sequence is represented as a point* $\boldsymbol{\theta}$ *on the simplex due to the constraints* $\theta_k \geq 0$ *and* $\sum_{k=1}^{K} \theta_k = 1$ *placed on the latent factors.*

distribution $P(\tilde{w} \mid \boldsymbol{\theta})$. From a geometric perspective, the *n*-gram sequence could now be represented as a point on the $(K-1)$-dimensional simplex, where the K vertices of the simplex are defined by the latent topics. Fig. 1 shows a 2-dimensional simplex for K = 3 latent topics.

3.2. Parameter estimation

We summarize the *expectation maximization* (EM) algorithm for learning the latent topics and inferring the latent factors as follows. In the E-step we infer the posterior probabilities of the latent factors $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_K\}$ for each of the *n*-gram sequences, $\mathcal{W} = (\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_t, \ldots, \tilde{w}_N)$. Exact inference is intractable, in which case we need to turn to variational method [10]. Let

$$q(\boldsymbol{\theta} \mid \boldsymbol{\gamma}) = \mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\gamma}) = C(\boldsymbol{\gamma}) \prod_{k=1}^{K} \theta_k^{\gamma_k - 1} \quad (7)$$

be the posterior probability of latent factors $\boldsymbol{\theta}$. The E-step consists of the following equations, where the parameters $\boldsymbol{\gamma} = \{\gamma_1, \gamma_2 \ldots, \gamma_K\}$ are updated:

$$\Phi(v,k) = \frac{\mathbf{B}(v,k)}{\sum_{k'=1}^{K} \mathbf{B}(v,k')} , \quad (8)$$

$$\gamma_k = \alpha_k + \sum_{v=1}^{V} \eta_v \cdot \Phi(v,k) , \; k = 1,2,\ldots,K . \quad (9)$$

In (8), the matrix

$$\mathbf{B}(v,k) = \mathbf{P}(v,k) \cdot \exp[\Psi(\gamma_k)] \quad (10)$$

and Φ are $V \times K$ matrices, and $\Psi(\cdot)$ is the digamma function. Recall that $\mathbf{P}$ is the $V \times K$ matrix containing the latent topics in its columns. In (9), η_v is the number of counts the term $\tilde{w}_v$ being observed in the sequence. For terms not observed in the sequence, in which case $\eta_v = 0$, the corresponding elements in the matrices Φ and $\mathbf{B}$ will be null and do not need to be computed or stored. This feature can be exploited to reduce the computation and memory requirement in the implementation. The only complication left is the computation of the digamma function $\Psi(\cdot)$ in (10), which can be resolved using standard statistical package. Notice that the denominator in (8) is just to ensure that Φ sum to one row wise. Equations (8) and (9) are iterated until convergence is met.

Once the posterior inference was done for all *n*-gram sequences, the latent topics $\mathbf{P}$ and the parameters $\boldsymbol{\alpha}$ of the Dirichlet prior are updated in the M-step as follows. Let D be the number of *n*-gram sequences available in the training data,

the new latent topics are computed by summing the Φ matrices over all sequences, as follows

$$\mathbf{P}(v,k) = \lambda_k \cdot \sum_{d=1}^{D} \eta_{v,d} \cdot \Phi_d(v,k) \,. \tag{11}$$

The normalization factor λ_k ensures that $\sum_{v=1}^{V} \mathbf{P}(v,k) = 1$ for each latent topic. The parameters $\boldsymbol{\alpha}$ of the Dirichlet prior are re-estimated with Newton-Raphson method using $\boldsymbol{\gamma}$ from the E-step as input. The E and M steps are repeated until convergence is met. Details of the Newton-Raphson method, convergence criterion and initialization of parameters can be found in [10].

3.3. Point estimates for latent factors

Using the latent topics trained from a sufficiently large corpus, we could analyze the decomposition of an unseen *n*-gram sequence into topics by looking at the posterior distribution in (7). To infer $q(\boldsymbol{\theta}|\boldsymbol{\gamma})$, we iterate between (8) and (9), where the parameters $\boldsymbol{\gamma} = \{\gamma_1, \gamma_2 \ldots, \gamma_K\}$ are updated until convergence.

The Dirichlet is a density function over the $(K-1)$-dimensional simplex. To better interpret the latent factors, one could use the point estimate:

$$\hat{\theta}_k = E_q\{\theta_k | \boldsymbol{\gamma}\} = \frac{\gamma_k}{\sum_{k'=1}^{K} \gamma_{k'}} \quad , \; k = 1,2,\ldots,K \,, \tag{12}$$

which gives the mean of the posterior distribution. This is different from the maximum *a posteriori* (MAP) criterion where the mode (i.e., the maximum) is used as the point estimate. The reason is that the mode may not exist when the latent factors become sparse in which only a few topics are responsible for generating the sequence. For the mode to exist we need $\gamma_k > 1, \forall k$. Notice also, if we let $\alpha_k = 0$ in (9), then (12) reduces to the ML estimate.

4. Language recognition in the topic simplex

The point estimate represents an *n*-gram sequence as a *K*-dimensional vector, which provides a more compact representation compared to the *V*-dimensional distribution over *n*-grams in (2). Using either form of compact or raw representations, the *n*-gram sequence is essentially mapped on to a simplex. To use these as inputs to SVM, we introduce two kernel metrics based on the Bhattacharyya measure which has shown to work well on the simplex [14]. Let $\mathcal{W}_a$ and $\mathcal{W}_b$ be two *n*-gram sequences, we could measure their similarity in the *V*-dimensional simplex as

$$\kappa_V(\mathcal{W}_a, \mathcal{W}_b) = \sum_{v=1}^{V} \sqrt{P_a(\tilde{w}_v) \cdot P_b(\tilde{w}_v)} \,. \tag{13}$$

For the *K*-dimensional topic simplex proposed in this paper, we use the following kernel metric:

$$\kappa_\theta(\mathcal{W}_a, \mathcal{W}_b) = \sum_{k=1}^{K} \sqrt{\hat{\theta}_{k,a} \cdot \hat{\theta}_{k,b}} \,. \tag{14}$$

Let κ denotes any of the kernel metric in (13) or (14), given $\mathcal{W}$ as input, the discriminant function of an SVM can be expressed as

$$f(\mathcal{W}) = \sum_{l=1}^{L} \pi_l y_l \kappa(\mathcal{W}_l, \mathcal{W}) + b \,, \tag{15}$$

where L is the number of support vectors, π_l are the weights assigned to the lth support vector with its label given by $y_l \in \{-1, +1\}$ and b is the basis parameter.

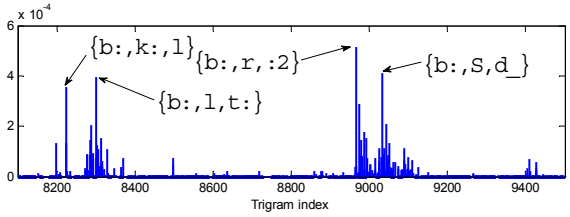

Figure 2: *A latent topic is a discrete distribution over n-grams. Indicated in the figure are four trigrams with high probabilities.*

For ease of implementation using standard SVM packages (e.g., *libSVM* or *SVMTorch*), the square-root operator is first applied element-wise on the inputs. SVM training can then be implemented using standard SVM packages with a linear kernel. In particular, we train one SVM for each language using the *one-versus-all* strategy [6].

5. Experiments

Experiments were carried out based on the NIST LRE07 closed-set language detection protocol [15]. There are fourteen target languages, which includes Arabic, Bengali, Chinese (comprised of Mandarin and three dialects), English, Farsi, German, Hindustani (comprised of Hindi and Urdu), Japanese, Korean, Russian, Spanish, Tamil, Thai and Vietnamese. We used CallFriend, OHSU, and some additional training data supplied by NIST/LDC to cover all target languages. For the phone recognizer, we used the Hungarian recognizer developed by Brno University of Technology [5]. The phone recognizer had been trained on the SpeechDat-East database to give 59 phones (and 3 non-phonetic units). Trigram counts were generated from the phone lattice [4]. Trigram with very low *inverse document frequency* (IDF) [16] were discarded, which leads to the final $V = 134{,}819$.

5.1. Latent topics

Unlike text documents, whereby latent topics can be literally understood [10], the latent topics derived from *n*-gram sequences are much obscured from intuitive interpretation. For text documents, there could have latent topics with specific themes referring to *Arts* or *Budget* with words including {*film*, *show*, *music*, *actress*,…} and {*million*, *tax*, *money*, *program*,…}, respectively. Fig. 2 shows a plot of latent topic arbitrarily selected from a set of $K = 50$. Indicated in the figure are trigrams with high probabilities in the topic. Though the latent topic could not be interpreted literally, we could still see that the four trigrams indicated in the figure exhibit similar pattern in which they all start with the same label `b:`.

Instead of looking at the interpretation of individual topics, the question that relates more to language recognition is how these topics represent individual languages. Fig. 3 shows two languages (i.e., English and Chinese) represented in terms of the distribution over the latent topics. The latent topics were trained using the development data as detailed in previous section. We then infer the topic proportions (i.e., the distribution over the latent topics) for individual language by iterating between (8) and (9) until convergence, and computing the topic proportion using (12). In this regard, a non-overlapping subset was selected from the training data before it was used for training the latent topics.

We deliberately arrange the topic indices so that three distinct groups can be seen in Fig. 3. (We make sure that the same set of indices is used when plotting the two distributions). Clearly, the topics on the right and left sides of the figure correspond more to English and Chinese

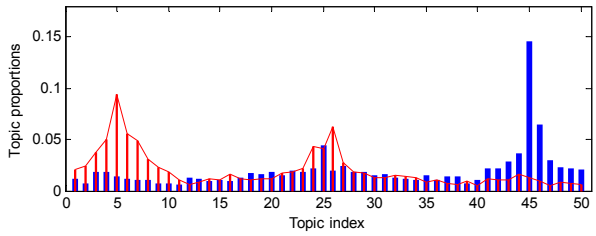

Figure 3: *Decomposing English (thick blue line) and Chinese (thin red line) languages into K = 50 topics.*

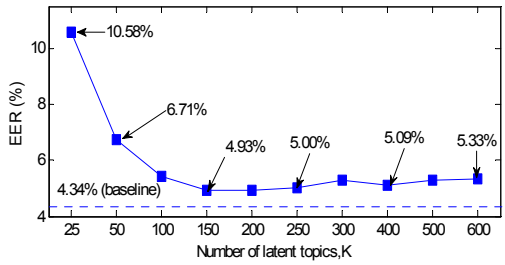

Figure 4: *EER at different number of latent topics K.*

respectively, while those at the middle are topics common to both. The remaining topics that fall within these groups are less significant in characterizing both languages. We observed almost the same pattern for different language pairs. This shows that language cues are preserved (each language has its own dedicated topics with some overlap between languages) by just using $K = 50$ topics, which is less than 0.05% of the original dimensionality of $V = 134{,}819$.

5.2. Language recognition

We evaluate the performance in terms of the average equal-error-rate (EER) computed from the pooled set of scores. Score normalization is performed using a simple back-end: $s_i' = s_i - \log\left(1/(T-1)\sum_{j\neq i} e^{s_j}\right)$, where $s_1, s_2, \ldots, s_T$ are the scores from the T target languages for a given test segment.

We used the PR-SVM with the kernel metric in (13) as the baseline system, while the kernel metric in (14) is used for the latent factors. Fig. 4 shows the EER evaluated at different values of K with a maximum at 600, which is less than 0.5% of the original dimensionality of $V = 134{,}819$. The EER decreases with an increasing number of latent topics K; however, the performance improvement levels off at $K = 150$, and get higher gradually at larger K. This probably due to the fact that we have only 2625 speech samples available for training the latent topics. The latent topics were not properly trained due to lack of data thereby introduce unnecessary ambiguity to the topic simplex.

Despite their low dimensionality, language recognition in the topic simplex shows competitive performance. At $K = 150$, we used only 0.11% of the dimensionality of the baseline system to obtain an EER of 4.93%. This gives an 11.97% difference compared to the baseline EER of 4.34%. Though this result does not match our expectation to surpass the performance of the baseline system, it does indicate that the latent factors preserve much of the language cues with a very small number of parameters. Furthermore, the low dimensionality allows additional processing (e.g., channel compensation) which we anticipate could further improve the performance.

6. Conclusions and future work

This paper has introduced and evaluated the use of latent topic modeling for spoken language recognition. The central idea is to constrain the variability of n-gram distribution within a low dimensional simplex of latent topics. To this end, we treat each n-gram as a discrete event, and represent an n-gram sequence on the topic simplex using the point estimate of the Dirichlet posterior. Language recognition using $K = 150$ latent factors (being only 0.11% of the original dimensionality of $V = 134{,}819$) results in an EER very close to that of the baseline. This result shows that the topic simplex could sufficiently capture the phonotactic cues pertaining to individual languages using a very low dimensionality. Future work will exploit the low dimensional benefit of the topic simplex to further improve the performance. In particular, we could use a full covariance for more effective channel compensation, which is difficult in the original space given the high dimensionality. We hope that this first attempt in using topic modeling would path the way to a framework similar to the *joint factor analysis* (JFA) [11] but for discrete count-based features.

7. References

[1] Y. K. Muthusamy, E. Barnard, and R. A. Cole, "Reviewing automatic language recognition," *IEEE Signal Process. Mag.*, vol. 11, no. 4, pp. 33-41, Oct. 1994.

[2] C. Chelba, T. Hazen, and M. Saraclar, "Retrieval and browsing of spoken content," *IEEE Signal Process. Mag.*, vol. 25, no. 3, pp. 39-49, May 2008.

[3] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 1, pp. 31-44, Jan. 1996.

[4] J. L. Gauvain, A. Messaoudi, and H. Schwenk, "Language recognition using phone lattices," in *Proc. Interspeech*, 2004, pp. 1215-1218.

[5] P. Matějka, P. Schwarz, J. Černocký, and P. Chytil, "Phonotactic language identification using high quality phoneme recognition," in *Proc. Eurospeech*, 2005, pp. 2237-2240.

[6] D. A. Reynolds, W. M. Campbell, W. Shen, and E. Singer, "Automatic language recognition via spectral and token based approaches," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, Ed. Berlin: Springer-Verlag, 2008.

[7] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, Upper Saddle River, New Jersey: Prentice Hall, 2000.

[8] C. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to ad hoc information retrieval," in *Proc. ACM SIGIR*, 2001, pp. 334-342.

[9] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning*, vol. 42, pp. 177-196, 2001.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, Jan. 2003.

[11] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, P. Dumouchel, "A study of inter-speaker variability in speaker verification," *IEEE Trans. Audio Speech and Language Processing*, vol. 16, no. 5, pp. 980-988, July 2008.

[12] D. Povey, "A tutorial-style introduction to subspace Gaussian mixture models for speech recognition," *Microsoft Research technical report MSR-TR-2009-111*, 2009.

[13] J.-T. Chien and C.-H. Chueh, "Dirichlet class language models for speech recognition," *IEEE Trans. Audio Speech Language Process.*, vol. 19, no. 3, pp. 482-495, Mar. 2011.

[14] K. A. Lee, C. H. You, H. Li, T. Kinnunen, and K. C. Sim, "Using discrete probabilities with Bhattacharyya measure for SVM based speaker recognition," *IEEE Trans. Audio Speech Language Process.*, accepted.

[15] *The 2007 NIST Language Recognition Evaluation Plan (LRE07)*, National Institute of Standards and Technology, July 2007.

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