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 investigates the use of latent topic modeling for spoken language recognition, where a topic is defined as discrete distribution over phone *n*-grams. These latent topics are trained in an unsupervised manner following the latent Dirichlet allocation (LDA) approach. We analyzed and observed that language cues can be represented very accurately in terms of the latent topics, where some latent topics are language specific while others exhibit multilingual characteristic. We then show that the latent topics can be used to define a low dimensional simplex (i.e., a bounded linear manifold) where language recognition can be done effectively. Experiments carried on the NIST 2007 language detection tasks show promising results.

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 indexing [2] to deal with immense amount of spoken documents without language labels.

State-of-the-art spoken language recognition systems use either phonotactic or acoustic cues. 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]. 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 [4]. This paper is concern 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 $M = 40$ phonetic units will lead to $V = M^3 = 64{,}000$ unique trigrams. This resulted in a very sparse estimate of *n*-gram probability with a lot of unseen *n*-grams, especially for the case of short segments, say in the range from three to ten seconds. 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.

In this paper, we assume that a latent factor model is responsible for generating the sequence of *n*-grams. The latent factor model consists of a set of latent topics, each being a *V*-dimensional distribution over *n*-grams. These latent topics define a lower dimensional simplex that confines the variability observed in the data, thereby reducing the number of parameters in the representing the observations. A simplex is a lifted and bounded hyperplane that does not go through the origin (see Fig. 1). Discrete distributions are confined to a simplex since their probabilities always sum to one.

Our aim in this paper is twofold: (i) to discover the hidden phonotactic constraints pertaining to individual languages and those common between languages in terms of latent topics, (ii) to find a low dimensional topic simplex for which spoken language recognition can be done effectively, especially for the case of short test segments. To this end, we use the latent Dirichlet allocation (LDA) approach proposed in [5] originally for modeling the word occurrence frequency in text documents. We show how LDA could be used to model *n*-gram sequence for the purpose of 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) [4] back-end. The front-end uses a phone recognizer[1] to convert speech waveform $\mathcal{X}$ into phone sequence:

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

where $P(Y \mid \mathcal{X}, \theta)$ denotes the posterior probability of generating the phone sequence Y given the input $\mathcal{X}$ and 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 time 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 is the phone set. Let $C\left(w_{t-n+1}^{t-1}, w_t\right)$ be the *n*-gram counts, the maximum likelihood (ML) estimate of the *n*-gram probability [6] can be computed as

$$P\left(w_{t-n+1}^{t-1}, w_t\right) = \frac{C\left(w_{t-n+1}^{t-1}, w_t\right)}{N} , \tag{2}$$

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 probability in (2) is a joint probability of n sub-events. This is slightly different from the LM approach, where 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 is modeled as conditional probability, $P(w_t \mid w_{t-n+1}^{t-1})$, in which the probability is conditioned on the preceding $(n-1)$ symbols (w_{t-n+1}^{t-1}). In this paper, the *n*-gram probabilities serve as inputs to SVM, for which earlier study [4] has shown that joint probabilities *n*-gram model is more suitable.

Furthermore, the joint probability (2) can be treated just like a unigram probability by letting $\tilde{w}_t = (w_{t-n+1}^{t-1}, w_t)$. The symbol $\tilde{w}_t$ now represents any of the $V = M^n$ possible unique *n*-grams, where M is the number of unique phones. By shifting one phone symbol at a time, a phone sequence is thereby converted into a sequence of *n*-grams. This is desirable as latent Dirichlet allocation (LDA) originally proposed in [5] works on unigram probability over words from text documents. Though it is possible to use LDA on conditional probability [7], 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 [5] for modeling the hidden topical structure of text documents. Since text documents are essentially sequences of words, similar technique can be applied to model *n*-gram sequences by treating the *n*-gram symbols as words in text documents.

3.1. Discovering the hidden structure in spoken languages

Let $\tilde{w}_t$ be any of the *V* possible *n*-grams. Following the LDA framework [5], we assume that an *n*-gram sequence $\mathcal{W} = (\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_t, \ldots, \tilde{w}_N)$, of length *N*, is generated by the following generative process:

(a) Draw a sample $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_K\}$ from a *K*-dimensional Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha})$ with parameters $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_K\}$.

(b) For each of the *N* symbols, draw $\tilde{w}_t$ from

$$P(\tilde{w}_t \mid \boldsymbol{\theta}, \beta_{1:K}) = \sum_{k=1}^{K} P(\tilde{w}_t \mid \beta_k) \times \theta_k \ . \tag{3}$$

Notice that the distribution $P(\tilde{w} \mid \boldsymbol{\theta}, \beta_{1:K})$ in (3) is obtained by combining *K* number of bases $P(\tilde{w} \mid \beta_k)$, each being a *V*-dimensional discrete distribution over all the possible *n*-gram symbols. We refer to these distributions as the *latent topics*. The weights $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_K\}$ that determine the proportions of topics in the mixture are called the *latent factors* (i.e., the hidden variables). The term *latent* is used as opposed to the *observable* nature of the variable $\tilde{w}_t$, which represents the *n*-gram symbols in our case. Once the topic proportions θ_k are determined as in (a), an *n*-gram sequence is generated by repeatedly sampling the same distribution as given in (3). Since the latent factors θ_k have to be positive and sum to one, the easiest choice of prior density is the Dirichlet distribution:

$$\mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) = C(\boldsymbol{\alpha}) \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{4}$$

where $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_K\}$ are the set of positive parameters and $C(\boldsymbol{\alpha})$ is the normalization factor ensuring that (4) is a legitimate density function.

By using latent factor model as described above, our aim is to discover the hidden phonotactic strands or cues underlying a particular language. We are also interested in learning the hidden structure and relationship between languages in terms of the latent topics. We could also

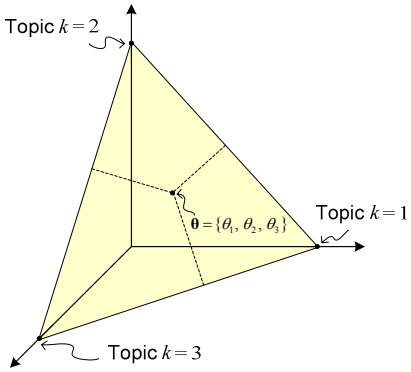

Figure 1: *A 2-dimesional simplex. At the vertices are the three latent topics. An n-gram sequence is represented as a point* $\boldsymbol{\theta}$ *on the simplex due to the constraints* $\theta_k \geq 0$ *and* $\sum_{k=1}^{3} \theta_k = 1$ *placed on the latent factors.*

decompose an *n*-gram distribution into combination of latent topics as in (3). 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 original *V*-dimensional *n*-gram distribution as given by the left-hand-side of (3). 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 the latent topics. This reduction in the number of parameters is of particular interest for the case of short test segment in language recognition. Fig. 1 shows a 2-dimensional simplex for $K = 3$ latent topics.

3.2. Parameter estimation

Parameters estimation is a reversal to the generative process described earlier in previous section. We summarize the expectation maximization (EM) algorithm for fitting the latent topics, Dirichlet prior and inferring 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 [5]. Let

$$q(\boldsymbol{\theta} \mid \gamma_{1:K}) = \mathrm{Dir}(\boldsymbol{\theta} \mid \gamma_{1:K}) = C(\boldsymbol{\alpha}) \prod_{k=1}^{K} \theta_k^{\gamma_k - 1} \tag{5}$$

be the approximate posterior probability of latent factors $\boldsymbol{\theta}$. The E-step consists of the following pair of update equations, where the parameters $\gamma_{1:K}$ are updated iteratively:

$$\Phi(v,k) = \frac{\mathbf{B}(v,k)}{\sum_{k'=1}^{K} \mathbf{B}(v,k')}, \tag{6}$$

$$\gamma_k = \alpha_k + \sum_{v=1}^{V} \eta_v \cdot \Phi(v,k) \ , \ k = 1,2,\ldots,K \ . \tag{7}$$

In (6), the matrix

$$\mathbf{B}(v,k) = \beta_k(v) \cdot \exp[\Psi(\gamma_k)] \tag{8}$$

and Φ are $V \times K$ matrices, and η_v is the number of counts of the term $\tilde{w}_v$ being observed in the sequence. These matrices are usually sparse depending on the length *N* of *n*-gram 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

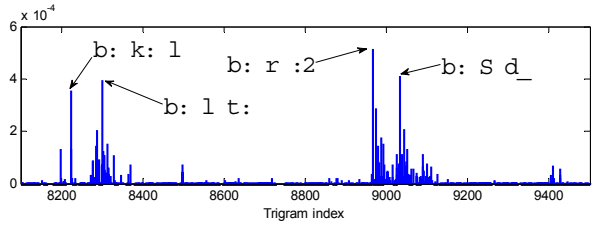

Figure 2: *A latent topic is a discrete distribution over n-grams. Indicated in the figure are four trigrams with high probabilities.*

memory requirement in the implementation. The only complication left is the computation of the digamma function $\Psi(\gamma_k)$ in (8), which can be resolved using standard statistical package. Equations (6) and (7) are iterated until convergence is met. Notice that the denominator in (6) ensures that Φ sum to one row wise.

Once the posterior inference is done for all *n*-gram sequences, the latent topics $\beta_{1:K}$ and the parameters $\boldsymbol{\alpha}$ of the Dirichlet prior are updated using the sufficient statistic obtained in the E-step. Let D be the number of *n*-gram sequences available in the training data, the new latent topics are computed by summing over the Φ matrices, as follows

$$\beta_k(v) = \lambda_k \cdot \sum_{d=1}^{D} \eta_{v,d} \cdot \Phi_d(v,k) . \tag{9}$$

The normalization factor λ_k ensures that $\sum_{v=1}^{V} \beta_k(v) = 1$ for each latent topics β_k for $k = 1,2,\ldots,K$. The parameters $\boldsymbol{\alpha}$ of the Dirichlet prior are re-estimated with Newton-Raphson method using $\gamma_{1:K}$ from the E-step as input. The E and M steps are iterated until convergence is met. Details of the Newton-Raphson method, convergence criterion and initialization of parameters can be found in [5].

3.3. Point estimates for latent factors

Using the latent topics $\beta_{1:K}(v)$ 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 (5). To infer $q(\boldsymbol{\theta}|\gamma_{1:K})$, we iterate between (6) and (7), where the parameters $\gamma_{1:K}$ are updated until convergence.

The Dirichlet is a density function over the $(K-1)$-dimensional simplex (Fig.1 shows a 2-dimensional simplex). To better interpret the latent factors, one could use the point estimate:

$$\hat{\theta}_k = E_q\{\theta_k | \gamma_{1:K}\} = \frac{\gamma_k}{\sum_{k'=1}^{K} \gamma_{k'}} \quad , k = 1,2,\ldots,K , \tag{10}$$

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 (7), then (10) reduces to the ML estimate.

4. Language recognition in the topic simplex

Using the point estimate $\hat{\boldsymbol{\theta}}$ in (10), we could now represent an *n*-gram sequence as a *K*-dimensional distribution over the latent topics, which provides a more compact representation

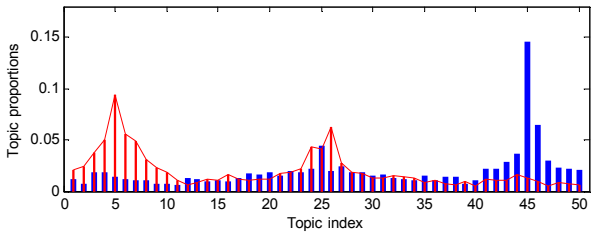

Figure 3: *Decomposing English (blue and thick line) and Chinese (red and thin line) languages into K = 50 topics.*

compared to the *V*-dimensional distribution over *n*-grams in (2). Using either form of compact or raw representation, the *n*-gram sequence is essentially mapped on to a simplex. To use this as input to SVM, we introduce two kernel metrics based on the Bhattacharyya measure which has shown to work well on the simplex [8]. Let $\mathcal{W}_a$ and $\mathcal{W}_b$ be two *n*-gram sequences, we could measure their similarity in the *V*-dimensional simplex as

$$\kappa_V(\mathcal{W}_a, \mathcal{W}_b) = \sum_{v=1}^{V} \sqrt{P_a(\tilde{w}_v) \cdot P_b(\tilde{w}_v)} , \tag{11}$$

or in the *K*-dimensional topic simplex as

$$\kappa_\theta(\mathcal{W}_a, \mathcal{W}_b) = \sum_{v=1}^{K} \sqrt{\hat{\theta}_{k,a} \cdot \hat{\theta}_{k,b}} . \tag{12}$$

Let κ denotes any of the kernel metric in (11) or (12), given $\mathcal{W}$ as input, the discriminant function of an SVM can be expressed as

$$f(\mathcal{W}) = \sum_{l=1}^{L} \pi_l y_l \kappa(\mathcal{W}_l, \mathcal{W}) + b , \tag{13}$$

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.

For ease of implementation using standard SVM packages (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 or exponential kernels. In particular, we train one SVM for each language using one-versus-less strategy [4].

5. Experiments

Experiments were carried out based on the NIST LRE07 closed-set language detection protocol [9]. 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 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 [10]. 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 [11]. Trigram with very low inverse document frequency (IDF) [12] were discarded, which leads to the final $V = 134{,}819$.

5.1. Latent topics

Unlike text documents, whereby latent topics can be literally understood [5], 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. Clearly, the latent topic in Fig. 2 could not be interpreted literally.

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 (6) and (7) until convergence, and computing the topic proportion using (10). 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 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 equal-error-rate (EER) computed from the pooled set of scores. The priors of the languages are equalized so that they have equal contributions to the EER [9]. Furthermore, score normalization is performed using a simple back-end: $s_i' = s_i - \log\left(1/T \sum_{j \neq i} e^{s_j}\right)$, where $s_1, s_2, \ldots, s_T$ are the scores from T target languages for a given test segment.

We used the PR-SVM with the kernel metric in (11) as the baseline system, while the kernel metric in (12) is used for the latent factors. Table 1 shows the EERs comparison under three test durations: 30 s, 10 s, and 3 s. Fig. 4 shows the EER evolution for different value of K with a maximum at 600, which is less than 0.5% of the original dimensionality of $V = 134{,}819$.

Despite its low dimensionality, the EER using the latent factors as feature are very close to that of the baseline. Though this 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. One possible reason for this is due to the robustness of SVM in handling high dimensional data. Nonetheless,

6. Conclusions

This paper has presented a simple approach in using latent topic modeling for spoken language recognition. The central idea is to constrain the variability of *n*-gram distributions within a low dimensional simplex of latent topics. To this end, we showed how the latent Dirichlet allocation (LDA) proposed originally for modeling text documents can be used to model the latent topics from *n*-gram sequences. Our study shows that the latent topics capture the phonotactic cues pertaining to individual languages and also those common between languages. Despite its low dimensionality (being only 0.5% of the original dimensionality), language recognition in the topic simplex resulted in EERs very close to that of the baseline.

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] 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.

[5] 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.

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

[7] 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.

[8] 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.

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

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

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

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