LANGUAGE RECOGNITION SYSTEM USING LANGUAGE BRANCH DISCRIMINATIVE INFORMATION

Xianliang Wang, Yulong Wan, Lin Yang, Ruohua Zhou, and Yonghong Yan

Key laboratory of Speech Acoustics and Content Understanding
Institute of Acoustics, Chinese Academy of Sciences
Beijing, 100190 China

ABSTRACT

This paper presents our study of using language branch discriminative information effectively for language recognition. Language branch variability (LBV) method based on factor analysis techniques is proposed. In LBV method, language branch variability factor is obtained by concatenating low-dimensional factors in the language branch variability spaces. Language models are trained within language branches and between languages. Experiments on NIST 2011 Language Recognition Evaluation (LRE) 30s, 10s and 03s tasks show the proposed LBV method provides stable improvement compared to the state-of-art total variability (TV) approach. In 30-second task, it gains relative improvement by 14.6% in equal error rate (EER) and 12.9% in minimum decision cost value (minDCF), and in new metrics of NIST 2011 LRE, it leads to relative improvement of 7.2%-17.7%.

Index Terms**—** language branch variability, discriminative information, factor analysis, language recognition.

1. INTRODUCTION

Language recognition aims to determine the language identity given a segment of speech. Two representative approaches have been widely used in language recognition, which are based on phonotactic features and spectral features. Phonotactic features attempt to extract high-level word syntactic information, and spectral features are based on the low-level acoustic signal properties [1]. Gaussian mixture model (GMM) [2] and support vector machine (SVM) [3] have been the choice in acoustic feature system over the past few years, gradually outperforming the phonetic system [4].

Many language recognition techniques have been developed based on GMM and SVM classifiers such as total variability. Total variability based on factor analysis has provided significant improvements to language recognition systems [5]. It maps a sequence of speech frames represented by an adapted GMM mean supervector onto a low-dimension total variability space. In the space, speaker and channel variabilities are contained simultaneously. Low-dimensional total factors (i-vectors) make it convenient to apply SVM classifier. Since SVM is a two-class classifier, one-vs-rest strategy is used to train classifiers for all languages.

With the development of language recognition, it is more concerned about the discrimination between the pairs of languages as is emphasized in the NIST 2011 Language Recognition Evaluation (LRE) [6]. In NIST 2011 LRE, more confusable target languages were evaluated, and new performance metrics which considered only the N worst performance language pairs were defined. This means that all the target languages should be modeled suitably and the discriminative ability between confusable pairs is more important.

In this paper, language branch variability (LBV) based on factor analysis is proposed. The LBV method aims to strengthen the discrimination between confusable languages. Languages can be divided into different language branches in perspective of linguistics. Languages in the same language branch share a remarkably similar pattern, and may be related through descent from a common ancestor, or be different dialects in a region. The proposed method considers the discriminative information of languages from both intra language branch and inter language branches in factor level and model level. In factor level, language branch variability factors are obtained by combing factors mapped on the language branch spaces. In model level, two groups of SVM models are trained. One group of models covers richer discriminative information of languages of the inner language branched and the other group emphasizes the discrimination between language branches. Experimental results show the discernment between confusable language pairs is stronger compared to the traditional factor analysis methods.

This paper is organized as follows: In Section 2, we give a simple review of SVM and total variability. Section 3 shows the proposed LBV approach in detail. Experimental setup and results are presented in Section 4. Finally, we conclude in

This work is partially supported by the National Natural Science Foundation of China (Nos. 10925419, 90920302, 61072124, 11074275, 11161140319, 91120001), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant Nos. XDA06030100, XDA06030500), the National 863 Program (No. 2012AA012503) and the CAS Priority Deployment Project (No. KGZD-EW-103-2).

Section 5.

2. BACKGROUND

2.1. Support Vector Machine

SVM is a popular technique for discriminative classification. The best separator of a SVM is defined by a kernel function as follows:

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i t_i K(\mathbf{x}, \mathbf{x}_i) + b \tag{1}$$

where N is the number of support vectors, α_i and b are the SVM parameters during the training step, and t_i is the label of the support vectors $\mathbf{x}_i$. The value of the label is either 1 or -1, depending upon whether the corresponding support vector belongs to class 1 or -1.

The kernel function $K(\cdot, \cdot)$ can be expressed as

$$K(\mathbf{x}, \mathbf{x}_i) = \phi(\mathbf{x})^{'} \phi(\mathbf{x}_i) \tag{2}$$

where $\phi(\mathbf{x})$ is a mapping from the input space to a possibly infinite dimensional SVM expansion space.

2.2. Total Variability Language Recognition

The total variability approach has become the state-of-art in both speaker verification and language recognition. It defines a new low-dimensional space mapping the high-dimensional GMM supervector to a low-dimensional and length-fixed vector.

For a given utterance, the language and variability dependent supervector is denoted as Eq. 3.

$$\mathbf{M} = \mathbf{m} + \mathbf{T}\mathbf{w} \tag{3}$$

where $\mathbf{m}$ is the supervector from Universal Background Model (UBM), $\mathbf{T}$ is language total variability space, and $\mathbf{w}$ is a standard-normally distributed latent variable called language total factor.

3. LANGUAGE RECOGNITION USING LANGUAGE BRANCH DISCRIMINANT INFORMATION

3.1. Language Branch

A language family is a group of languages related to each other through descended from a common ancestor in historical linguistics. Language branch is established by languages sharing common features that other languages are not found in the common ancestor of the language family, for example, West Slavic Branch and East Slavic Branch, which are from the Common Slavic Family.

Membership of languages in the same language branch is established by comparative linguistics, genetically relative or contact languages. Two languages are considered to be genetically relative if one is descended from the other as Hindi and Urdu or the two languages are descended from a common ancestor as Czech and Slovak. A language may also be contacted with other languages by influencing each other, for example, language transferring and specifically borrowing as Japanese and Chinese. Mixed languages, pidgin languages and creole languages also construct language branch for their inseparable relationship.

3.2. Language Branch Variability Method

The NIST 2011 LRE differed from the previous ones in emphasizing the language pair condition, and it contained more confusable language pairs such as Hindi/Urdu, Czech/Slovak and Lao/Thai and so on. The most confusable pairs were generally within clusters of linguistically similar languages.

Total variability has been introduced to language recognition and obtained significantly improved performance. But the total variability doesn't emphasize the discrimination of confusable pairs. In this paper, we introduce the idea of language branch to language recognition, and we called it language branch variability method.

The LBV method pays more attention to the discriminative information of languages between and within language branches in both factor level and model level.

In factor level, language branch variability factors are extracted by mapping the GMM supervector onto all of the language branch variability spaces while the total variability onto the total variability space. For a given utterance, the language and variability dependent supervector is denoted in Eq. 4.

$$\mathbf{M} = \mathbf{m}_{b_i} + \mathbf{T}_{b_i} \mathbf{w}_{b_i} \tag{4}$$

where b_i represents language branch i, $\mathbf{m}_{b_i}$ is the GMM supervector of language branch i, $\mathbf{T}_{b_i}$ is language branch variability space and $\mathbf{w}_{b_i}$ is variability factor vector in the space. For simply, the following GMM of this paper represents a specific language branch GMM.

The process of training language branch variability space of a language branch is exactly the same as learning total variability space, except for the GMM supervector instead of UBM supervector. The language branch variability factor is obtained by concatenating factors in all of the language branch spaces as follows:

$$\mathbf{w} = [\mathbf{w}_{b_1}, \mathbf{w}_{b_2}, ..., \mathbf{w}_{b_i}, ..., \mathbf{w}_{b_L}] \tag{5}$$

where L is the number of language branches.

In this way, different variability factors within the same language branch represent different characteristics of different utterances, and variability factors of all the language branches constitute the language branch variability factor covering richer discriminative information between language branches.

The variability factor of language branch i is defined as follows:

$$\mathbf{w}_{b_i} = (\mathbf{I} + \mathbf{T}^t_{b_i}\Sigma^{-1}{}_{b_i}\mathbf{N}_{b_i}(u)\mathbf{T}_{bi})^{-1}\mathbf{T}^t{}_{b_i}\Sigma^{-1}{}_{b_i}\hat{\mathbf{F}}_{b_i}(u) \quad (6)$$

We define $\mathbf{N}_{b_i}(u)$ as a diagonal matrix whose diagonal blocks are $N_{bi}{}^{[c]}\mathbf{I}$. $\hat{\mathbf{F}}_{b_i}(u)$ is obtained by concatenating all the first-order Baum-welch statistics $F_{bi}{}^{[c]}$ for utterance u. Σ_{b_i} is a common diagonal covariance of GMM. $N_{bi}{}^{[c]}$ and $F_{bi}{}^{[c]}$ can be obtained as follows:

$$N_{bi}{}^{[c]} = \sum_{t=1}^{T} P(c|y_t, \Omega_{b_i}) \quad (7)$$

$$F_{bi}{}^{[c]} = \sum_{t=1}^{T} P(c|y_t, \Omega_{b_i})(y_t - m_{b_i}{}^{[c]}) \quad (8)$$

where T is the number of frames, c is the Gaussian index and Ω_{b_i} is the diagonal covariance matrix estimated as [7]. $P(c|y_t, \Omega_{b_i})$ corresponds to posterior probability of mixture component c generating the vector y_t. $m_{b_i}{}^{[c]}$ is the mean of GMM mixture component c.

Two groups of SVM models are trained in LBV. Scores of the two groups of models are fused as the final results. A block diagram of the process is shown in Figure 1 taking Arabic Iraqi as example.

In the first group, models are trained within language branch, and we emphasize the discrimination of languages within language branch. Here we take Arabic Iraqi as example. We assigned Arabic Iraqi, Arabic Levantine, Arabic Maghrebi, Arabic MSA to Arabic Branch. In the process of training Arabic Iraqi model, utterances of Arabic Iraqi are seen as positive samples and utterances of the other languages in Arabic Branch are the negative samples. In this process, discrimination of confusable language pairs are more prominent.

As for the second group of models, target language models are trained among all the languages. The positive samples of target language are composed of utterances of the language, and the utterances of all the other languages merged to the negative samples. Language models in this group cover the discriminative information of languages of different language branches.

4. EXPERIMENTS

4.1. Corpora and Evaluation

Our Experiments were carried out on the NIST LRE 2011 closed-set task. There were 24 target languages in corpora of 2011 evaluation database: Arabic Iraqi, Arabic Levantine, Arabic Maghrebi, Arabic MSA, Bengali, Czech, Dari, English American, English India, Farsi/Persian, Hindi, Lao, Mandarin, Panjabi, Pashto, Polish, Russian, Slovak, Spanish, Tamil, Thai, Turkish, Ukrainian and Urdu. Equal error rate (EER) and the minimum decision cost value (minDCF) were used as old metrics [8] for evaluation. We also used minimum and actual average cost value ($Cavg$) as new metrics for evaluation [6], in terms of: the new $Cavg$ computed on the 24 language-pairs with the highest $minCavg$ ($minNCavg$, $actNCavg$) and the $Cavg$ computed for all 276 language-pairs ($minFCavg$, $actFCavg$).

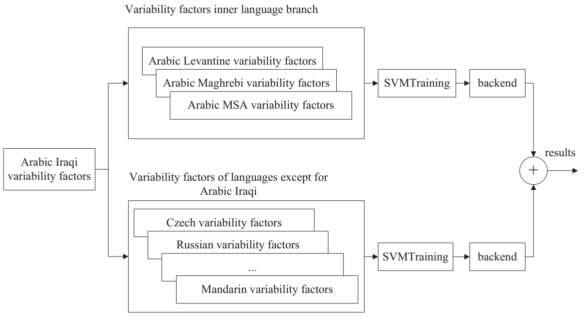
Variability factors inner language branch
Arabic Levantine variability factors
Arabic Maghrebi variability factors
Arabic MSA variability factors
SVMTraining
backend
Arabic Iraqi variability factors
results
+
Variability factors of languages except for Arabic Iraqi
Czech variability factors
Russian variability factors
...
Mandarin variability factors
SVMTraining
backend

Fig. 1. A block diagram for training two groups of SVM models, taking Arabic Iraqi as example

Table 1. Classification of the 24 target languages in NIST 2011 LRE according to language branches

Language Name	**Language Branch**
Czech, Slovak, Polish	West Slavic Branch
Russian,Ukrainian	East Slavic Branch
Bengali, Hindi, Panjabi, Urdu	Indic Branch
Farsi/Persian, Pashto, Dari	Iranian Branch
Arabic Iraqi, Arabic Levantine, Arabic Maghrebi, Arabic MSA	Arabic Branch
English American, English India	English Branch
Mandarin, Lao, Thai	Sino-Tibetan Branch
Spanish	Spanish Branch
Turkish	Turkish Branch
Tamil	Tamil Branch

4.2. Experiment Setup

The 24 target languages in NIST 2011 LRE were divided into ten language branches according to the linguistic knowledge as Table 1.

Our experiments were operated on the Mel Shifted Delta Coefficients (MSDC) feature [9], with 7 Mel Frequency Cepstral Coefficients (MFCC) concatenated with Shifted Delta Coefficients (SDC) 7-1-3-7 feature. Features were normalized to mean 0 and variance 1 using Cepstral Mean Subtract (CMS) and Cepstral Variance Normalization (CVN). UBM

Table 2. Results of the total variability (TV) and language branch variability (LBV) methods in EER(%) and minDCF(%).

	30s		10s		03s	
system	EER	minDCF	EER	minDCF	EER	minDCF
TV	7.19	7.46	14.34	14.55	29.26	29.33
LBV	6.14	6.50	13.02	13.31	28.80	28.79

Table 3. Results of the total variability (TV) and language branch variability (LBV) methods in $minNCavg$, $actNCavg$, $minFCavg$ and $actFCavg$.

	system	minNCavg	actNCavg	minFCavg	actFCavg
30s	TV	0.1259	0.1581	0.0271	0.0392
	LBV	0.1159	0.1467	0.0223	0.0335
10s	TV	0.2277	0.2482	0.0811	0.0927
	LBV	0.2172	0.2370	0.0723	0.0836
03s	TV	0.3439	0.3577	0.2178	0.2293
	LBV	0.3408	0.3527	0.2119	0.2228

and GMMs used in our experiments contained 1024 Gaussians. The dimension of total factors was 400. SVMTorch [10] with a linear inner-product kernel function was implemented to train the one-vs-rest SVM classifier. LDA and diagonal covariance Gaussians backend [11] are used to calculate the log-likelihoods for target languages.

4.3. Experimental Results

Results of the total variability and the proposed LBV language recognition system on NIST 2011 LRE are presented. EER and minDCF of 30s, 10s and 03s tasks are observed in Table 2. Table 3 shows results of the new evaluation metrics on 30s, 10s and 03s tasks. DET plots of these two systems are shown in Figure 2 for test durations 30s. Figure 3 gives the performance of the two systems in terms of minDCF, and the 15 language pairs in the same language branch from the 24 worst language pairs are observed.

In Table 2, it is shown that the LBV method performs consistently better than the total variability language recognition system. On 30s task, it achieves a relatively reduction of 14.6% in EER and 12.9% in minDCF. And the LBV method improves relatively by 9.3% in EER and 8.5% in minDCF compared to the total variability system on 10s task.

NIST LRE 2011 focus on difficult to distinguish language pairs and utilized new evaluation metrics. Table 3 shows that the LBV method improved by 7.9%, 7.2%, 17.7% and 14.5% on $minNCavg$, $actNCavg$, $minFCavg$, and $actFCavg$ respectively compared to the total variability system on 30s task and 4.6%, 4.5%, 10.9% and 9.8% on 10s task.

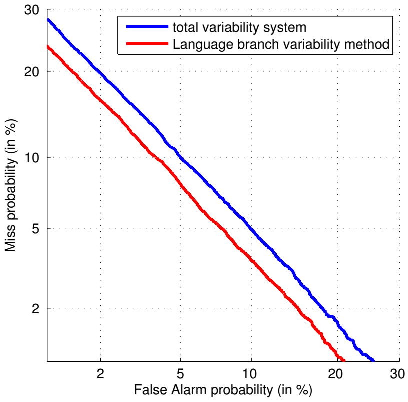

Fig. 2. DET curves of systems on NIST 2011 LRE 30s task.

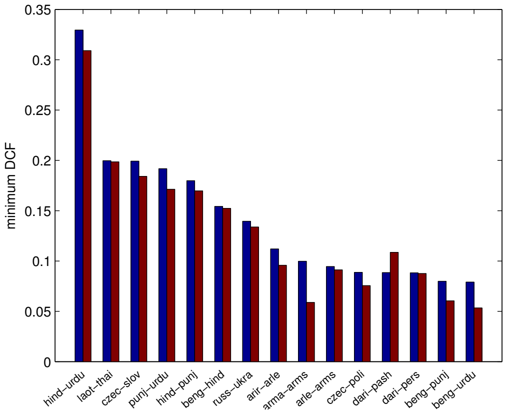

Fig. 3. minDCF of the total variability and LBV for 15 language pairs which are in the same language branches from the 24-worst pairs on NIST 2011 LRE 30s task.

In Figure 3, the 15 worst language pairs shared by total variability and LBV system are observed, which achieve varying degree reduction compared to the total variability except for Dari/Pashto pairs in minDCF.

5. CONCLUSION

In this paper, a novel language recognition system named LBV based on factor analysis is proposed. The LBV method introduces the knowledge of linguistics to language recognition. Variability factor vectors extracted in all of the language branch spaces are concatenated and constitute language branch variability factor. Language models are trained within language branches and between languages. Experiments show that the proposed LBV method outperforms the total variability system significantly. Future work may include exploring new approaches based on language branch to improve performance of language recognition.

6. REFERENCES

[1] Haipeng Wang, Cheung-Chi Leung, Tan Lee, Bin Ma, and Haizhou Li, “Shifted-delta mlp features for spoken language recognition,” *IEEE Signal Processing Letters*, vol. 20, pp. 15–18, 2013.

[2] Lukas Burget, Pavel Matejka, and Jan Cernocky, “Discriminative training techniques for acoustic language identification,” in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*. IEEE, 2006, vol. 1.

[3] William M Campbell, Joseph P Campbell, Douglas A Reynolds, Elliot Singer, and Pedro A Torres-Carrasquillo, “Support vector machines for speaker and language recognition,” *Computer Speech & Language*, vol. 20, no. 2, pp. 210–229, 2006.

[4] Najim Dehak, Pedro A Torres-Carrasquillo, Douglas A Reynolds, and Reda Dehak, “Language recognition via i-vectors and dimensionality reduction.,” in *INTERSPEECH*, 2011, pp. 857–860.

[5] David Martınez, Oldrich Plchot, Lukás Burget, Ondrej Glembek, and Pavel Matejka, “Language recognition in ivectors space,” *Proceedings of Interspeech, Firenze, Italy*, pp. 861–864, 2011.

[6] Craig S Greenberg, Alvin F Martin, and Mark A Przybocki, “The 2011 nist language recognition evaluation.,” in *INTERSPEECH*, 2012.

[7] Patrick Kenny, Gilles Boulianne, and Pierre Dumouchel, “Eigenvoice modeling with sparse training data,” *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 3, pp. 345–354, 2005.

[8] Alvin F Martin and Mark A Przybocki, “Nist 2003 language recognition evaluation,” in *Proceedings of Eurospeech*. Geneva, Switzerland: Sept, 2003, vol. 1344.

[9] Pedro A Torres-Carrasquillo, Elliot Singer, Mary A Kohler, Richard J Greene, Douglas A Reynolds, and John R Deller Jr, “Approaches to language identification using gaussian mixture models and shifted delta cepstral features.,” in *INTERSPEECH*, 2002.

[10] Ronan Collobert and Samy Bengio, “Svmtorch: Support vector machines for large-scale regression problems,” *The Journal of Machine Learning Research*, vol. 1, pp. 143–160, 2001.

[11] Mohamed Faouzi BenZeghiba, Jean-Luc Gauvain, and Lori Lamel, “Language score calibration using adapted gaussian back-end.,” in *INTERSPEECH*, 2009, pp. 2191–2194.