EXEMPLAR BASED LANGUAGE RECOGNITION METHOD FOR SHORT-DURATION SPEECH SEGMENTS

Meng-Ge Wang, Yan Song, Bing Jiang, Li-Rong Dai, Ian McLoulghlin

University of Science and Technology of China, Department of Electronic and Engineering, Hefei, Anhui, China, 230027

ABSTRACT

This paper proposes a novel exemplar-based language recognition method for short duration speech segments. It is known that language identity is a kind of weak information that can be deduced from the speech content. For short duration speech segments, the limited content also leads to a large intra-language variability. To address this issue, we propose a new method. This borrows a vector quantization based representation from image classification methods, and constructs the exemplar space using the popular i-vector representation of short duration speech segments. A mapping function is then defined to build the new representation. To evaluate the effectiveness of our proposed method, we conduct extensive experiments on the NIST LRE2007 dataset. The experimental results demonstrate improved performance for short duration speech segments.

Index Terms—Language Recognition, i-vector, Vector Quantization

1. INTRODUCTION

Recently, i-vector based language recognition systems have achieved *state-of-the-art* performance in the NIST language recognition evaluations [1, 2]. Unlike in JFA (Joint Factor Analysis), the low-dimension i-vector representation of a given speech utterance is obtained by projecting them onto a *total variability* space ***T***, which may circumvent the problem of distinguishing the additional information, such as channel and speaker *etc*. Traditionally, the ***T*** space is trained by pooling a large number of speech utterances from different languages.

Despite the success of i-vector based systems on long duration test utterances, performance on short duration test utterances, such as 10 seconds or 3 seconds, is still far from satisfactory [1, 2]. It is known that the language information is latent, which can be reflected by speech content. The i-vector can be considered to be a compressed representation of salient speech segment variability in a low dimensionality ***T*** space [3]. Long duration speech segments contain adequate content information, and the ***T*** space trained from long duration speech segments can thus effectively characterize the language information. However, short duration speech utterances, with their limited speech content, may lead to large intra-language variations. These are introduced through different speech content, different speakers, channels *etc*. These variations tend to distract the distribution of the resulting i-vector representation, and degrade performance.

Recently, some authors addressed the short duration test conditions in the related speaker recognition task. In [4], Sarkar *et.al*. studied the effect of i-vector modeling on short and mismatched utterance duration for speaker verification. They proposed training several ***T*** spaces using speech segments with different durations, and then fuse the scores obtained from each ***T*** space for the sake of higher performance. However, training separate ***T*** spaces, especially for short duration speech segments, is computational costly. Larcher *et. al.* [3] performed speaker recognition using short duration utterances with i-vectors but constrained the input to fixed lexical content. This constraint would hinder real-life application, especially in the language identification task.

In this paper, we investigate an efficient method of effectively enhancing performance on short duration test utterances. To achieve this goal, we borrow the vector quantization based representation framework from image classification research, and propose an exemplar-based representation of short utterances under the i-vector paradigm. Specifically, we first define an exemplar space for efficiently characterizing the variability of the short duration segments. This exemplar space consists of the templates obtained by unsupervised clustering from the duration matched i-vectors. We then define a mapping from the original i-vector space to the exemplar space and form a new representation. Finally, SVM is used to train the language classifier.

Compared with [4], our method takes advantage of computational efficiency, while still being able to tackle the duration mismatch issue. To evaluate the effectiveness of our proposed method, we conduct extensive experiments using the NIST LRE2007 dataset. The experimental results demonstrate the superiority of our proposed system.

The rest of the paper is organized as follows. Section 2 gives a brief review of the paradigm of i-vector system. In Section 3 we propose a duration-dependent dictionary method in total variability space which relatively improves the performance on 10s test conditions by about 12% compared with the state-of-the-art i-vector based system.

Experimental results and conclusion are given in sections 4 and 5 respectively.

2. I-VECTOR BASED LANGUAGE RECOGNITION FRAMEWORK

The i-vector technique provides an elegant way to transform a feature sequence into a fixed-length vector of low dimensionality while preserving most of the relative information among speech segments. First introduced for speaker recognition [5], the i-vector representation of a specific utterance is modeled as follows.

$$\boldsymbol{M} = \boldsymbol{m} + \boldsymbol{T}\boldsymbol{\omega} \quad (1)$$

where $\boldsymbol{m}$ indicates the stacked mean vector of UBM, $\boldsymbol{T}$ is the low-rank loading matrix whose columns are the basis spanning the total variability space, and $\boldsymbol{\omega}$ is a standard normal distributed hidden variable, termed the i-vector. Since there is no distinction between inter- and intra-class variability in this concept, post processing is necessary to remove nuisance effects. Typically, this involves LDA (Linear Discriminate Analysis) and WCCN (Within Class Covariance Normalization), leading to a final i-vector representation of

$$\boldsymbol{\omega}' = \boldsymbol{B}^T \boldsymbol{A}^T \boldsymbol{\omega} \quad (2)$$

where the $\boldsymbol{A}$ and $\boldsymbol{B}$ correspond to the LDA and WCCN projection matrices respectively. A simple but effective Cosine Similarity Scoring is adopted for classification. Meanwhile, the low dimensionality of i-vector features makes the employment of discriminative classifiers such as SVM very efficient.

In i-vector based language recognition methods, it is important to derive the total variability space $\boldsymbol{T}$. Traditionally, the $\boldsymbol{T}$ space is trained using long duration speech segments which are assumed to contain adequate information for modeling the latent language information. However, the T space may fail to capture the high variability that exists in short duration speech segments. To address this issue, we propose an exemplar-based representation for short duration speech utterance based on the i-vector.

3. EXEMPLAR-BASED LANGUAGE RECOGNITION METHOD FOR SHORT DURATION UTTERANCE

The proposed exemplar-based language recognition method for short duration utterances is shown in Figure 1. Its framework consists of two components, 1) Template learning based on exemplars, in which the templates are obtained by unsupervised clustering methods to give a better modeling of the distribution. 2) Representation of short duration utterances, in which a mapping function is defined to form a new representation that is capable of conveying a more comprehensive description of the correlation between speech representation and templates. In particular, the proposed framework is deployed under the total variability paradigm due to the low dimensionality of i-vectors, which would make the template learning and representation mapping procedures much more efficient. Moreover, rather than training loading matrices with respect to each test condition, as reported in [4], our approach exploits the existing loading matrix obtained from long duration utterance, termed $\boldsymbol{T}_{long}$, to derive the i-vector representation through equation (1).

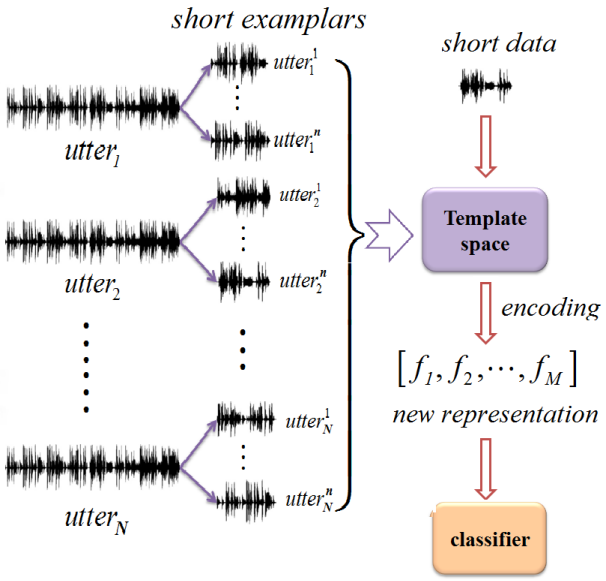

Figure 1. The exemplar based language recognition framework for short duration utterance

3.1. Exemplar-based template construction

The assumption that a template derived directly from duration-matched data can give a better description about the manifold on which the short-term test data resides underlies our template learning formulation. To fulfill the assumption, long-term training utterances are split into short segments whose actual speech duration is homologous to the test condition. I-vectors are then extracted by projecting these short utterances into the subspace defined by $\boldsymbol{T}_{long}$. The reason for discarding duration matched $\boldsymbol{T}_{short}$ is that even though the space on which short utterances characterize languages differs to that characterized by long data, the significant variability in content between two short segments would cause the loss of important discriminative language information in the extracted i-vectors.

A *K*-means algorithm is employed to construct our template from the short-term i-vectors due to its simplicity, efficiency and wide adoption in clustering based dictionary learning. This iterative algorithm incrementally improves the dispersion among clusters thereby better representing the exemplar space with the centroids obtained. More precisely, given a set of short-term training i-vectors labeled as the *i*-th

language, the corresponding sub-template $\boldsymbol{P}_i$ with K components can be obtained by partitioning these i-vectors into K clusters and then the centers $\{\mu_i^k\}_{k=1}^K$ are taken to be the descriptive vectors in the form of $\boldsymbol{P}_i=[\mu_i^1, \mu_i^2, ..., \mu_i^K]$. All the sub-templates will constitute the final dictionary as follow:

$$\boldsymbol{P} = \left[\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_L\right] \tag{3}$$

3.2. Representation of short duration utterances

The constructed template $\boldsymbol{P}$ is then used to map short duration i-vectors into new representations for the following classifier training and identification process. Several encoding algorithms are investigated in this paper, specifically the hard-alignment encoder, cosine encoder, soft-thresholding encoder and Gaussian encoder. New features are normalized into unit length before entering the next step.

3.2.1. Hard-alignment encoding

Inspired by the straightforward link between the K-means algorithm and vector quantization, we pair the hard-alignment encoding approach with our exemplars. In the vector quantization concept, each item of data is replaced by the closest component in the dictionary. However, in our method when the template $\boldsymbol{P}$ and a specific i-vector $\boldsymbol{\omega}$ are given, we search for the most similar exemplar to $\boldsymbol{\omega}$ in each sub-dictionary and take the cosine distance between them as the corresponding element in the new feature $\boldsymbol{f}$, while the others remain as zero.

3.2.2. Cosine encoding

As an analogue to the efficient Cosine Similarity Scoring in classical i-vector systems, the cosine encoder takes the cosine distances between the i-vector $\boldsymbol{\omega}$ and each atom in the template $\boldsymbol{P}$ to form the new representation. Namely, we take

$$\boldsymbol{f}_{\mathrm{i}} = \frac{\boldsymbol{\omega}^{\mathrm{T}} \boldsymbol{p}_{\mathrm{i}}}{\|\boldsymbol{\omega}\| \cdot \|\boldsymbol{p}_{\mathrm{i}}\|} \tag{4}$$

where $\boldsymbol{f}_i$ and $\boldsymbol{p}_i$ refer to the i-th component in the new feature and the template respectively.

3.2.3. Soft-threshold encoding

Soft-threshold is a simple but effective feed-forward encoding scheme with a preset threshold α that can be adjusted to match the dataset [6]. Each element in the new feature $\boldsymbol{f}$ is computed as:

$$\boldsymbol{f}_{\mathrm{i}} = \max\left\{0, \left|\boldsymbol{\omega}^{\boldsymbol{T}} \boldsymbol{p}_i\right| - \alpha\right\} \tag{5}$$

This activation function may be thought of as a simple form of "competition" between features, in which half of feature features will be set to 0. The threshold parameter α is determined using the 10-fold cross-validation on training set.

3.2.4. Gaussian encoding

Owing to the assumption of Gaussian distributed i-vectors we paired the dictionary with an encoder in conjunction with a Gaussian-shaped kernel [7]. Specifically, the i-th element of $\boldsymbol{f}$ is obtained by computing:

$$\boldsymbol{f}_{\mathrm{i}} = \exp\left\{-\frac{\|\boldsymbol{\omega} - \boldsymbol{p}_{\mathrm{i}}\|_2}{2\sigma^2}\right\} \tag{6}$$

where the scale parameter σ is set as the mean points-to – center distance in each cluster.

3.3. Classifiers

Further processing is implemented in the new feature space. After normalizing the new features into unit length, Support Vector Machine (SVM) based on cosine kernel with one-versus-all strategy is chosen as our classifier. And the distance from the test feature to a specific separating hyperplane is outputted as the score of this test sample against the corresponding language.

4. EXPERIMENT SETUP

The proposed duration-dependent dictionary approach is based on the NIST LRE2007 corpora which comprises 14 languages of conversational telephone speech (CTS) data.

Training and test utterances are converted into 56-dimensional shifted delta cepstra features followed by VAD and Gaussian normalization. The 1024-mixture UBM and loading matrix $\boldsymbol{T}$ representing a 400-dimensional total variability subspace are trained with 5766 samples of durations ranging from 2 to 4mins. Then we split the long-term utterances into 30s, 10s and 3s (actual speech length) respectively and get their SDC features. Together with the test features, i-vectors are extracted using the previously trained $\boldsymbol{T}$ to form the 30s-training, 10s-training, 3s-training and test dataset. Around 15000 i-vectors are randomly picked from the two training sets respectively to train the SVM classifiers. The rest of the data are used to generate the duration-dependent template. LDA and WCCN are applied to each i-vector to boost the discrimination among data of different languages while minimizing the intra-class variance.

5. RESULTS AND ANALYSIS

This section gives the results of the NIST LRE2007 close-set task in the 10s and 3s test conditions. The performance is evaluated in terms of Equal Error Rate (EER) without backend processing. Different encoding schemes are compared and analyzed in this section. Our baseline is formulated as the classical i-vector system based on Cosine Similarity Scoring.

Figure 2 illustrates the changes in performance of the cosine encoding scheme against the size of sub-dictionary increase. Neither conditioning process nor backend are applied in this case but we can still discern patterns in these

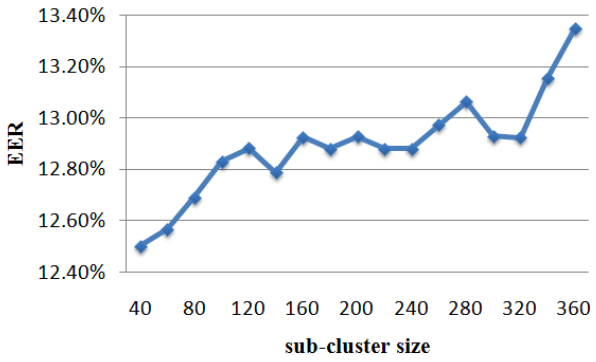

Fig. 2. The performance degradation with the growth of sub-dictionary size on 10s test data.

raw results. They show that large sub-cluster size does not yield better performance. This could be attributed to the increase in ambiguity when aligning an i-vector to a specific cluster if there are too many potential target clusters available. Consequently, and as a consideration of computational complexity, we constrain each sub-dictionary to a small size in the following experiments.

The comparisons of the system using our proposed exemplar-based representation with different encoding approaches are described in Section 3 are reported in Table 1. It can be clearly seen from the table that the hard-assignment encoding method did not work well with our system. This could be attributed to the fact that the 1-of-K coding scheme results in very crude quantization, which cannot provide sufficient discriminative information.

As alternatives to the vector quantization method, the cosine encoder and soft-threshold encoder (where the parameter α is set to 0.055) can achieve an improvement of around 10% on the 30s test task, and 13% on the 10s test task compared with the baseline. Moreover, as can be seen from the table, systems with the Gaussian encoding scheme give 13% and 16% performance gain for 30s and 10s test data respectively. Contrary to the significant improvement achieved on the 10s test condition, performance of the 3s test only increased by about 6% using the cosine and soft-threshold encoders. This could be ascribed to the lack of language information in a 3s segment. Overall, from the experimental results we can conclude that with an appropriate choice of mapping scheme our proposed exemplar-based representation of limited-duration utterances is effective in enhancing the LID performance on short test conditions especially in terms of 10s tasks.

6. CONCLUSION AND FUTURE WORK

In this paper, we proposed an exemplar-based representation of short duration utterances for language identification using i-vectors. Our proposed method can be considered as a practical and efficient way to reduce the effects of duration mismatch directly in the classical total variability subspace system. This representation can yield consistently better performance for both 10s and 3s test conditions over existing state-of-the-art systems. In addition, the effectiveness of several encoding schemes is discussed and compared, showing that the simpler cosine encoder achieves the best performance.

Encoding method	30s EER(%)	10s EER(%)	3s EER(%)
Baseline	4.39	10.89	22.38
Hard alignment	6.29	11.12	24.84
Cosine encoding	3.88	9.50	21.09
Soft threshold	3.93	9.45	**20.95**
Gaussian encoding	**3.84**	**9.08**	21.13

Table 1. Performance showed in EER with different encoding schemes on 30s, 10s and 3s test condition without backend. Our baseline is the classical i-vector system using Cosine Similarity Scoring.

In the future, we intend to continue the study on enhancing performance for short duration test conditions, specifically the 3s condition, by searching for more powerful learning algorithms and encoders, such as sparse coding [8] and deep belief networks [9]. Besides this, effective backends will be studied for the sake of yielding further performance improvement.

7. ACKNOWLEDGEMENT

This work was supported by the National Science Foundation of China (NSFC) under the Grant No. 61172158, and Chinese Universities Scientific Fund(CUSF) under grant No. Wk2100060008.

7. REFEENCES

[1] Najim Dehak, Pedro A. Torres-Carrasquillo, Douglas Reynolds, and Reda Dehak, "Language Recognition via i-vectors and Dimensionality Reduction," in *Proc. of INTERSPEECH*, Florence, Italy, 2011, 857-860.

[2] D. Martinez, O. Plchot, L. Burget, O. Glembek, and P. Matejka, "Language Recognition in iVector Space," in *Proc. of INTERSPEECH*, Florence, Italy, 2011, 861-864.

[3] A. Larcher, P.M. Bousquet, K.A. Lee, D. Matrouf, H. Li, and J.F. Bonastre, "I-vectors in the context of phonetically-constrained short utterances for speaker verification," in *Proc. of ICASSP*, Kyoto, Japan, 2012, 4773-4777.

[4] A.K. Sarkar, D. Matrouf, P.M. Bousquet, and J.F. Bonastre, "Study of the effect of i-vector modeling in short and mismatch utterance duration for speaker verification," in *Proc. of INTERSPEECH*, Portland, Oregon, 2012.

[5] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Trans. on Audio, Speech and Language Processing*, vol.19, pp. 788-798, 2011.
[6] A. Coates and A.Y. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proc. of the 28th ICML*, Bellevue, Washington, USA, 2011, 921-928.
[7] J.C. van Gemert, J.M. Geusebroek, C.J. Veeman and A.W.M Smeulders, "Kernel Codebooks for Scene Categorization," in *Proc. of ECCV*, Marseille, France, 2008, 696-709.
[8] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust Face Recognition via Sparse Representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.31, no.2, 2009, pp. 210-227.
[9] G.E. Hinton, S. Osindero and Y.W. The, "A Fast Learning Algorithm for Deep Belief Nwts," *Neural Computation, vol 18, no. 7*, 1527-1554.