International Journal of Speech Technology
https://doi.org/10.1007/s10772-017-9476-3

Sparse coding of i-vector/JFA latent vector over ensemble dictionaries for language identification systems

Om Prakash Singh[1] · Rohit Sinha[1]

Received: 18 July 2017 / Accepted: 10 November 2017

Abstract

Computing sparse representation (SR) over an exemplar dictionary is time consuming and computationally expensive for large dictionary size. This also requires huge memory requirement for saving the dictionary. In order to reduce the latency and to achieve some diversity, ensemble of exemplar dictionary based language identification (LID) system is explored. The full diversity can be obtained if each of the exemplar dictionary contains only one feature vector from each of the language class. To achieve full diversity, a large number of multiple dictionaries are required; thus needs to compute SR for a particular test utterance as many times. The other solution to reduce the latency is to use a learned dictionary. The dictionary may contain unequal number of dictionary atoms and it is not guaranteed that each language class information is present. It totally depends upon the number of data and its variations. Motivated by this, language specific dictionary is learned, and then concatenated to form a single learned dictionary. Furthermore, to overcome the problem of ensemble exemplar dictionary based LID system, we investigated the ensemble of learned-exemplar dictionary based LID system. The proposed approach achieves the same diversity and latency as that of ensemble exemplar dictionary with reduced number of learned dictionaries. The proposed techniques are applied on two spoken utterance representations: the i-vector and the JFA latent vector. The experiments are performed on 2007 NIST LRE, 2009 NIST LRE and AP17-OLR datasets in closed set condition.

Keywords Language identification · Joint factor analysis · i-Vector · Sparse representation · Exemplar dictionary · Learned dictionary · Learned-exemplar dictionary

1 Introduction

Language identification (LID) refers to the task of identifying the language of the unknown spoken discourse with help of machines. The industries like travel and tourism, telecommunication and emergency services typically require interface in multiple languages, thus employ human resource having working knowledge of multiple languages. With the expansion of such industries and the need for providing round the clock services, the cost of maintaining such services becomes exorbitantly high. To address this challenge, a number of automated multi-lingual services have been developed in the past. The success of such multi-lingual services depend on the fast and accurate determination of the spoken language by the LID system. Over the years, many LID approaches are developed exploiting acoustic, phonotactics and syntactical differences among languages. The reader is referred to Ambikairajah et al. (2011) for a concise survey of those methods.

This work deals with the acoustic-domain LID methods. Owing to broad similarity, often the techniques developed for speaker recognition also find applicability in the LID domain. The joint factor analysis (JFA) and the i-vector are the examples of this trend. Following the success of the i-vector paradigm in the speaker recognition domain, the same has been explored in the LID domain and forms the state-of-the-art acoustic-domain LID method (Martinez et al. 2011; Dehak et al. 2011; Jiang et al. 2012). In the current works, the language space is modeled by class-specific adaptation of the mean parameter of Gaussian mixture model based universal background model (GMM-UBM). In the i-vector approach, a low dimensional total variability space (Dehak

✉ Om Prakash Singh
o.singh@iitg.ernet.in

Rohit Sinha
rsinha@iitg.ernet.in

[1] Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, Guwahati 781039, India

Published online: 22 November 2017

et al. 2011) is defined that simultaneously accounts for both language and channel variability in the data. In addition to that, the JFA (Kenny et al. 2007) based LID approach is also commonly employed. In this, the high dimensional GMM mean-supervectors representation of the spoken utterances are analyzed into language and session/channel subspaces to achieve robustness in practical conditions.

In recent past, the sparse representation classification (SRC) has been successfully applied in various pattern recognition tasks such as face recognition (Wright et al. 2009), speaker identification (Naseem et al. 2010), speaker verification (Kua et al. 2011) and language identification (Jiang et al. 2012; Singh et al. 2013; Singh and Sinha 2017). In those works, the sparse coding obtained over an exemplar dictionary are employed in the classification. The exemplar dictionary is created by simply grouping the vector representations of class-specific training examples. With the increasing size of the exemplar dictionary, the computational burden of sparse coding over such dictionary becomes quite prohibitive. For addressing this problem, the *k*-means clustering and the random sampling are explored for the SRC based LID in Jiang et al. (2012).

In literature, for achieving more effective sparse coding many dictionary learning methods are reported (Aharon et al. 2006; Mairal et al. 2010). In context of speech based speaker verification, the K-SVD dictionary based sparse codings are reported to yield better detection performance than those obtained with an exemplar dictionary (Haris and Sinha 2012, 2015). In these works, the dictionaries are created using GMM mean-supervector based representation of the spoken utterances, so the cost of sparse coding is quite high.

Motivated by that, in this work we propose sparse coding of two low dimensional features (i-vectors and JFA latent vector) over an ensemble derived from either simple-exemplar or learned-exemplar dictionaries for developing the LID system. For dictionary learning, K-SVD (Aharon et al. 2006) and online dictionary learning (ODL) (Mairal et al. 2010) algorithms are investigated. We hypothesize that such an ensemble of dictionaries is expected to provide the diversity gain. The different subsets of the development data are being created by taking a fixed number of utterances from the data pool for each of the languages. Any two subsets consist of entirely different utterances, and hence orthogonal subsets are created from the development data covering all the languages in the task. Our proposed ensemble of exemplar dictionary design is different than multiple exemplar dictionary reported in Jiang et al. (2012). The proposed approach doesn't require any random sampling and hence there is no chance of getting same feature vector in different dictionaries (i.e., no repetition of selecting the same feature vector). The best performance of the proposed ensemble dictionary based LID system is noted if each of exemplar dictionary contains only one feature vector from each of the languages (i.e., each dictionary size being equal to the number of languages). The number of such an exemplar dictionaries would become very high if the size of the developmental dataset is large. On account that many sparse codings of the test feature vector, the computational burden of the proposed approach becomes prohibitive. In order to mitigate this problem, we further explore the ensemble of learned exemplar dictionaries based LID system. It is expected that the proposed ensemble of learned-exemplar dictionary based LID approach performs efficiently for large size development data, where both dictionary learning and diversity can be utilized properly. A separate cosine distance score (CDS) obtained for each of the dictionaries in the ensemble SR based systems are then fused to produce the final decision score. The performances of the developed systems are contrasted with the state-of-the-art i-vector and the JFA latent vector based LID system employing within class covariance normalization (WCCN) (Hatch et al. 2006) for session/channel compensation and various classifiers (CDS, SRC, support vector machine (SVM), linear generative Gaussian, regularized multi-class logistic regression (MLR) and Gaussian-PLDA). All the proposed LID systems are explained by considering i-vector as the spoken utterance representation. Similar steps can be used when the JFA latent vector is chosen as the utterance representation.

The rest of the paper is organized as follows: Sect. 2 describes the i-vector/JFA latent vector based contrast systems employing various classifiers. The recent phonetic temporal neural (PTN) based LID system is also developed for contrast purpose. The proposed ensemble of exemplar dictionary based LID with SR is presented in Sect. 3. The proposed LID approach using SR over learned-exemplar dictionary is presented in Sect. 4. The experimental setup is described in Sect. 5, followed by the results and discussion in Sect. 5.5. Finally the paper is concluded in Sect. 9.

2 Contrast systems

2.1 i-vector and the JFA latent vector based LID systems

In this section, i-vector and the JFA latent vector based LID systems are developed for the contrast purpose. In the i-vector framework, the basic idea is to captures both language and session/channel variability jointly by constructing a low rank projection matrix referred to as total variability matrix ($\boldsymbol{T}$-matrix). The $\boldsymbol{T}$-matrix is learned following the procedure described in Kenny et al. (2005) and requires a large amount of developmental data for estimating it properly. Given the $\boldsymbol{T}$-matrix, a language and session/channel dependent GMM mean-supervector $\boldsymbol{s}$ is modeled as,

$$\boldsymbol{s} = \boldsymbol{m} + \boldsymbol{T}\boldsymbol{w} \tag{1}$$

where $\boldsymbol{m}$ is the language independent UBM mean-supervector and $\boldsymbol{w}$ is low dimensional representation of GMM mean-supervector derived using factor analysis. The vector $\boldsymbol{w}$ is popularly referred to as *identity vector* (i-vector).

Unlike the i-vector extraction, in the JFA case both language and session/channel variabilities are modeled separately. For this purpose, a language and session/channel dependent GMM mean-supervector $\boldsymbol{s}$ is modeled as,

$$\boldsymbol{s} = \boldsymbol{m} + \boldsymbol{V}\boldsymbol{v} + \boldsymbol{D}\boldsymbol{d} + \boldsymbol{U}\boldsymbol{u} \tag{2}$$

where $\boldsymbol{m}$ is language and session independent UBM mean-supervector. The matrices $\boldsymbol{V}$ and $\boldsymbol{D}$ are eigenvoice matrix and diagonal residue matrix, respectively and jointly define the language subspace. The matrix $\boldsymbol{U}$ is eigenchannel matrix and defines the session/channel subspace. The vectors $\boldsymbol{v}$, $\boldsymbol{d}$ and $\boldsymbol{u}$ are the language and session/channel dependent factors in their respective subspaces. In our implementation, we have estimated eigenvoice matrix $\boldsymbol{V}$ and eigenchannel matrix $\boldsymbol{U}$ from the developmental dataset, ignoring $\boldsymbol{D}$. The estimated language vector $\boldsymbol{v}$ is used as the utterance representation and is referred to as *JFA latent vector* in this work.

2.2 LID system based on the i-vectors and SRC

Assume there are L distinct languages in the training set with lth language containing n_l example utterances. Let $\boldsymbol{\omega}_{lj} \in R^m$ denotes m-dimensional jth i-vector representation for the lth language, where $l = 1, 2, \dots, L$ and $j = 1, 2, \dots, n_l$ denote the indices of the languages and the training i-vectors in the lth language, respectively. It is assumed that a target i-vector $\boldsymbol{y}$ belonging to the lth language class can be approximated as,

$$\boldsymbol{y} \approx \gamma_{l1}\boldsymbol{w}_{l1} + \gamma_{l2}\boldsymbol{w}_{l2} + \dots + \gamma_{ln_l}\boldsymbol{w}_{ln_l} \tag{3}$$

where $\{\gamma_{lj}\}$ are the real scalar coefficients.

For the SRC purpose, an exemplar dictionary $\boldsymbol{E}$ is formed by stacking the sub-matrices corresponding to L languages as,

$$\boldsymbol{E} = [\boldsymbol{E}_1, \boldsymbol{E}_2, \dots, \boldsymbol{E}_L] \in R^{m \times n}, \quad n = \sum_{l=1}^{L} n_l \tag{4}$$

where the lth matrix $\boldsymbol{E}_l = [\boldsymbol{w}_{l1}, \boldsymbol{w}_{l2}, \dots, \boldsymbol{w}_{ln_l}] \in R^{m \times n_l}$ is formed by stacking all training i-vectors corresponding to that language. As $m \ll n$, the resultant dictionary $\boldsymbol{E}$ turns out to be overcomplete. Note that the dimensionality of the i-vector happens to be much smaller than the total number of examples available for all the languages in a typical database.

A target i-vector $\boldsymbol{y}$ can be represented as linear combination of a few columns of the exemplar dictionary $\boldsymbol{E}$ as

$$\boldsymbol{y} \approx \boldsymbol{E}\boldsymbol{x} \tag{5}$$

where $\boldsymbol{x} \in R^n$ is the *sparse vector (s-vector)* of unknown coefficients. The orthogonal matching pursuit (OMP) algorithm (Pati et al. 1993) can be used to find the solution to (5) as,

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} ||\boldsymbol{y} - \boldsymbol{E}\boldsymbol{x}||_2 \quad \text{subject to } ||\boldsymbol{x}||_0 \leq \gamma \tag{6}$$

where γ is the chosen constraint on the sparsity controlling the number of nonzero coefficients in the s-vector $\hat{\boldsymbol{x}}$.

The elastic net (ENet) (Zou and Hastie 2005) based sparse coding which uses both l_1 and l_2 penalty terms can be also used for computing the SR, and is given by

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}||\boldsymbol{y} - \boldsymbol{E}\boldsymbol{x}||_2 + \lambda_1||\boldsymbol{x}||_1 + \lambda_2||\boldsymbol{x}||_2^2 \tag{7}$$

where λ_1 and λ_2 are the regularization parameters. The LASSO is a special case of ENet with $\lambda_2 = 0$.

In the s-vector $\hat{\boldsymbol{x}}$, ideally all nonzero coefficients should correspond to the dictionary columns (atoms) from the language class to which the target i-vector $\boldsymbol{y}$ belongs to. This is valid based on the assumption in (3). However, in practice, the s-vector does have some nonzero coefficients other than the class of $\boldsymbol{y}$ owing to modeling error, noise and session/channel variability. The score for identification is found by taking the mean of the language specific coefficients in the sparse vector $\hat{\boldsymbol{x}}$.

2.3 LID system based on the i-vectors and CDS

In contrast to using SRC, one can also compute the similarity between two i-vectors for language identification. Given the training and test i-vectors ($\boldsymbol{w}_{trn}$ and $\boldsymbol{w}_{tst}$), the cosine distance scoring (CDS) is computed as,

$$score(\boldsymbol{w}_{trn}, \boldsymbol{w}_{tst}) = \frac{(\boldsymbol{w}_{trn})^t \boldsymbol{w}_{tst}}{||\boldsymbol{w}_{trn}||_2 ||\boldsymbol{w}_{tst}||_2} \tag{8}$$

Using (8), the scores against all training i-vectors are are computed for a particular test i-vector. Finally, the class-wise mean of the scores are computed for the final decision.

2.4 LID system based on the i-vectors and SVM

The support vector machine (SVM) (Vapnik 2013) is a supervised binary classifier. In the training phase, a set of training instance-class label pairs $(\boldsymbol{w}_i, c_i), i = 1, 2, \dots, N, \boldsymbol{w}_i \in \mathbb{R}^m, c_i \in (-1, +1)$ is used to determine the best linear hyperplane H, which maximizes the margin between the two classes. The notations $\boldsymbol{w}_i$ and c_i represents ith i-vector and corresponding class label. The classification function f associated with the optimal hyperplane H is given by:

$f : \mathbb{R}^N \rightarrow \mathbb{R}$

$$w :\mapsto f(w) = \sum_{i=1}^{M} \alpha_i c_i k(\mathbf{w}, \mathbf{w}_i) + b \tag{9}$$

where α_i and b are the Lagrange multiplier and bias, the SVM parameters obtained from training step. Considering the linear kernel function $k(\mathbf{w}, \mathbf{w}_i) = \mathbf{w}_i^t \mathbf{w}$, the Eq. (9) can be re-written as

$$f(w) = \left(\sum_{i=1}^{M} \alpha_i c_i \mathbf{w}_i \right)^t \mathbf{w} + b = \mathbf{a}^t \mathbf{w} + b \tag{10}$$

where $\mathbf{a} = \left(\sum_{i=1}^{M} \alpha_i c_i \mathbf{w}_i \right)$ is the weight vector. For the given test i-vector $\mathbf{w}_{tst}$, the classification is performed by noting the sign of the function $f(\mathbf{w}_{tst})$. The 1-vs-all strategy is used to obtain the scores for all languages.

2.5 LID system based on the i-vectors and generative Gaussian model

In the linear generative Gaussian (GG) model (Martinez et al. 2011), language specific i-vectors are used to train a language dependent multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}_l, \boldsymbol{\Sigma})$, where full covariance matrix $\boldsymbol{\Sigma}$ is shared across all languages. Given the test i-vector $\mathbf{w}$, the log-likelihood score for each language is computed as

$$\ln p(\mathbf{w}|l) \simeq \mathbf{w}^t \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_l - \frac{1}{2} \boldsymbol{\mu}_l^t \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_l \tag{11}$$

where $\boldsymbol{\mu}_l$ is the mean vector and $\boldsymbol{\Sigma}$ is the common covariance matrix. Equation (11) is linear in $\mathbf{w}$, and hence lead to linear classifier.

2.6 LID system based on the i-vectors and logistic regression

Given a set of training instance-class label pairs $(\mathbf{w}_i, c_i), i = 1, 2, \dots, N$ where $\mathbf{w}_i \in \mathbb{R}^m$ and $c_i \in (0, 1)$ are the ith i-vector and class label respectively. The probability distribution of the class label c given an i-vector $\mathbf{w}$ can be modeled using logistic regression (Ng 2004), and given by

$$p(c = 1|\mathbf{w}; \boldsymbol{\theta}) = \sigma(\boldsymbol{\theta}^t \mathbf{w}) = \frac{1}{1 + exp(-\boldsymbol{\theta}^t \mathbf{w})} \tag{12}$$

where $\boldsymbol{\theta} \in \mathbb{R}^m$ are the parameters of logistic regression model and $\sigma(\cdot)$ is the sigmoid function. The regularized logistic regression model can be described as

$$\arg\max_{\theta} \sum_{i=1}^{N} logp(c_i|\mathbf{w}_i; \boldsymbol{\theta}) - \lambda R(\boldsymbol{\theta}) \tag{13}$$

where $R(\boldsymbol{\theta})$ is the regularization term. The parameter $\boldsymbol{\theta}$ can be computed using (13). The value of $R(\boldsymbol{\theta}) = ||\boldsymbol{\theta}||_2$ and $R(\boldsymbol{\theta}) = ||\boldsymbol{\theta}||_1$ corresponds to l_1 and l_2 regularized logistic regression. Once $\boldsymbol{\theta}$ is computed, find the decision boundary for the classification. The decision boundary is defined as the line where

$$p(c = 1|\mathbf{w}; \theta) = 0.5 \Longrightarrow \theta^t \mathbf{w} = 0 \tag{14}$$

For multi-class problem, 1-vs-all strategy is used to obtain the scores for all languages.

2.7 LID system based on the i-vectors and Gaussian PLDA

Assume $\mathbf{w}_k$ represents the kth i-vector of a language, where $k = 1, 2, \dots, K$. The Gaussian probabilistic linear discriminant analysis (G-PLDA) (Prince et al. 2007) model assumes that each i-vector $\mathbf{w}_k$ can be decomposed into language component $\mathbf{l}$ and channel component $\mathbf{c}$ and given by

$$\mathbf{w}_k = \mathbf{l} + \mathbf{c} = (\mathbf{m} + \boldsymbol{\Phi}\boldsymbol{\phi}) + (\boldsymbol{\Psi}\boldsymbol{\psi}_k + \boldsymbol{\eta}_k) \tag{15}$$

The language and channel component describes the between-language and within-language variability respectively. The former doesn't depend on the particular utterance while the later is utterance dependent. The columns of Φ and Ψ provides a basis for the language subspace (eigenvoice) and channel subspace (eigenchannel) respectively. The $\boldsymbol{\phi}$ and $\boldsymbol{\psi}_k$ are the corresponding latent identity vectors having standard normal distributions. The residual term $\boldsymbol{\eta}_k$ is assumed to be Gaussian with zero mean and diagonal covariance Σ. The global offset is represented by m. In this work, we have used the modified G-PLDA (Kenny 2010) model due to low dimensional i-vector. Here, eigenchannels have been removed and Σ is considered as full covariance matrix. The modified G-PLDA model is given by

$$\mathbf{w}_k = \mathbf{m} + \boldsymbol{\Phi}\boldsymbol{\phi} + \boldsymbol{\eta}_k \tag{16}$$

The EM algorithm (Dempster et al. 1977) is applied on the large development data for ML point estimates of the model parameters m, Φ, Σ.

2.8 Phonetic temporal neural

The PTN (Tang et al. 2017) approach is the combination of phonetic deep neural network (DNN) followed by temporal model. The phonetic representation (DNN phonetic feature) can be extracted from the output layer (phone posterior) or the last hidden layer (logits) and can be applied to temporal model (LSTM-RNN) to capture the phonetic temporal properties of a language with a high temporal resolution. The time delay neural network (TDNN) can be used for the extraction of DNN phonetic feature.

3 Proposed ensemble exemplar dictionary based LID system

The computational burden of sparse coding over an exemplar dictionary grows up with the increase of training examples. To address this issue, a single compact exemplar dictionary was created by Jiang et al. (2012) using two different dictionary construction methods (i.e., random sampling and k-means clustering). In random sampling approach, the language specific sub-dictionary was created by randomly selecting the training examples from the particular language. Then, sub-dictionaries were concatenated to form a single compact exemplar dictionary. In this process, it is guaranteed that some of the training examples are un-utilized. For fixed database, the number of un-utilized training examples increases with decrease of sub-dictionary size. To some extent, authors (Jiang et al. 2012) have tried to resolve the issue with random subspace method (Ho 1998), where multiple exemplar dictionaries were created using random sampling approach, and scores were averaged to get the final decision. By doing this, there is a probability of selecting the same training examples. This probability is large especially in limited database. Apart from this, issues related to optimal choice of dictionary size and the number of dictionaries required to achieve the maximum diversity gain were not addressed.

In order to investigate these issues, we propose ensemble exemplar dictionary approach, which not only reduces the computational cost in sparse coding but may enhance the LID performance due to diversity in sparse coding of the target. The proposed approach doesn't require any random sampling and hence there is no chance of getting same feature vector in different dictionaries (i.e., no repetition of selecting the same feature vector).

Figure 1 shows the flow diagram of the proposed ensemble exemplar dictionary based LID system. The salient steps involved in the proposed system are summarized below:

1. Given a pool of m-dimensional WCCN session/channel compensated developmental i-vectors corresponding to each of the L languages in the task.
2. Create an exemplar dictionary by selecting k unique i-vectors from each of L languages. Let G be the number of such exemplar dictionaries get created.
3. For each of G exemplar dictionaries
   (a) Find the sparse code (i.e., the s-vector) of the training and test i-vectors.
   (b) Compute the CDS between the train and test s-vectors.
4. For final decision, fuse the resulting G scores using multi-class logistic regression.

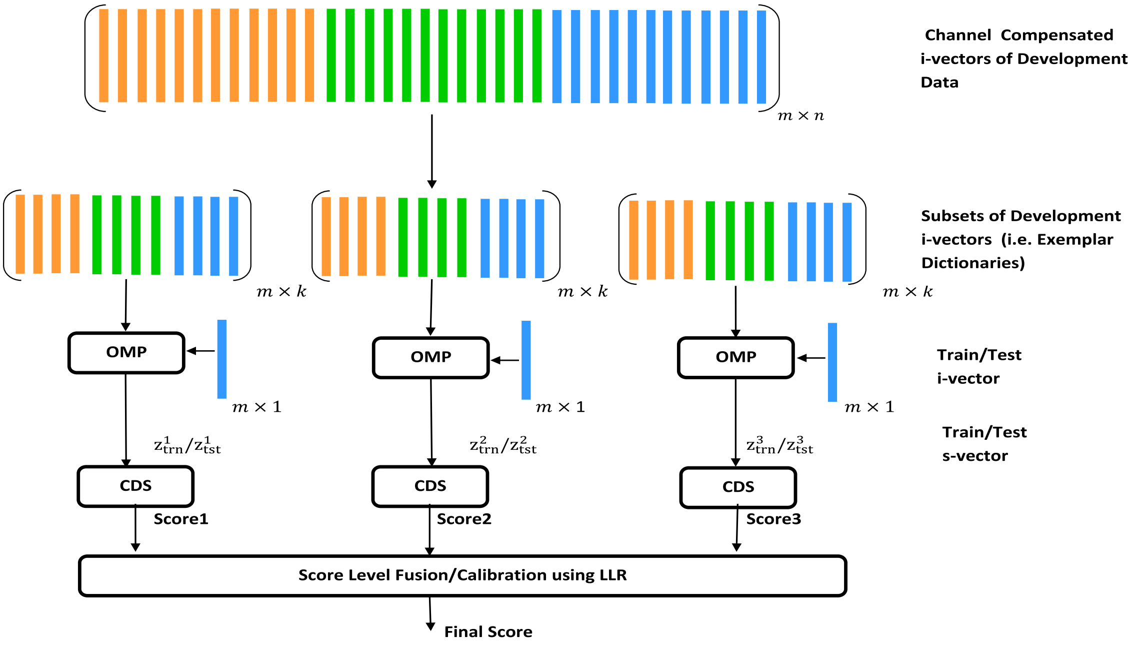

Fig. 1 Flow diagram of the proposed ensemble exemplar dictionary based LID approach. For ease of illustration, the session/channel compensated i-vectors for three language classes are depicted in orange, green and blue colors. The sparse codes (i.e., the s-vectors) of the train and test i-vectors over ensemble of the exemplar dictionaries are computed and denoted by z^1_{trn}/z^1_{tst}, z^2_{trn}/z^2_{tst} and z^3_{trn}/z^3_{tst}, each of size $k \times 1$. The CDS of each pair of the train and test s-vectors is computed and fused using linear logistic regression (LLR) for the language identification. (Color figure online)

4 Proposed learned-exemplar dictionary based LID approach

The ensemble exemplar dictionary based LID approach presented in Sect. 3 can become cumbersome when a large number of examples per language are available. Towards addressing this problem, we explored compression of number of examples involved in creation of exemplar dictionary using the existing dictionary learning techniques. In this work, we have made use of the K-SVD and ODL algorithms.

4.1 Single learned-exemplar dictionary based LID System

Given the developmental data for the lth language containing P numbers of m-dimensional i-vectors $\boldsymbol{W}^{\boldsymbol{l}} = \{\boldsymbol{w}^{\boldsymbol{l}}{}_i\}_{i=1}^{P}$ and the constraint on sparsity as γ', the problem of dictionary learning can be defined as,

$$\underset{\boldsymbol{D}^l, \boldsymbol{Z}^l}{\arg\min} \parallel \boldsymbol{W}^{\boldsymbol{l}} - \boldsymbol{D}^{\boldsymbol{l}}\boldsymbol{Z}^{\boldsymbol{l}} \parallel_2^2 \quad \text{subject to} \quad \parallel \boldsymbol{z}^{\boldsymbol{l}}{}_i \parallel_0 \leq \gamma' \tag{17}$$

where $\boldsymbol{D}^{\boldsymbol{l}}$ is the language specific learned dictionary having k columns and $l = 1, \dots, L$. The matrix $\boldsymbol{Z}^{\boldsymbol{l}}$ denotes a set of sparse vectors corresponding to $\boldsymbol{W}^{\boldsymbol{l}}$ with $\boldsymbol{z}^{\boldsymbol{l}}{}_i$ being the sparse vector for the ith i-vector of the lth language. All language specific dictionaries are estimated and then combined to form a single learned-exemplar dictionary as,

$$\boldsymbol{D} = [\boldsymbol{D}^{\boldsymbol{1}} \mid \boldsymbol{D}^{\boldsymbol{2}} \mid, \dots, \mid \boldsymbol{D}^{\boldsymbol{L}}] \tag{18}$$

where '|' denotes a horizontal concatenation operator.

Once the dictionary $\boldsymbol{D}$ is obtained, the i-vector $\boldsymbol{w}$ for an utterance is sparse coded using the OMP or elastic net algorithm (ENet) (Zou and Hastie 2005). The OMP based sparse coding problem is formulated as

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\arg\min} \parallel \boldsymbol{D\alpha} - \boldsymbol{w} \parallel_2 \quad \text{subject to} \quad \parallel \boldsymbol{\alpha} \parallel_0 \leq \gamma'' \tag{19}$$

where γ'' is the chosen constraint of the sparsity. However, the ENet based sparse coding problem is formulated as

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\arg\min} \frac{1}{2} \parallel \boldsymbol{D\alpha} - \boldsymbol{w} \parallel_2 + \lambda_1 \parallel \boldsymbol{\alpha} \parallel_1 + \lambda_2 \parallel \boldsymbol{\alpha} \parallel_2^2 \tag{20}$$

The LASSO is a special case of ENet with $\lambda_2 = 0$.

For language detection, the s-vectors corresponding to training and test utterances are determined and the similarity score is computed using CDS. It is worth noting that one can also learn a single K-SVD dictionary combining examples of all the languages. But, unlike the one in (18), such a dictionary would be deficient in two aspects:

1. the class labels of the atoms are unknown,
2. the set atoms belonging to different classes in the dictionary may not be closely similar in size.

The first one only precludes SRC so CDS can be used. Whereas the second one is critical as it affects the sparsity constraint being kept same for all the classes.

4.2 Ensemble learned-exemplar dictionary based LID system

To exploit the diversity, an ensemble of smaller sized dictionaries are created by partitioning the language specific learned dictionaries in the manner already discussed in Sect. 2.2. Each of the language specific K-SVD learned dictionaries is designed to have same number of columns. Thus, unlike the simple exemplar dictionary, the size of derived learned-exemplar dictionary can be kept fixed even when more and more developmental data is made available. Further for the best LID performance, the multiple exemplar dictionaries in the ensemble approach are required to have only one atom per language. The size of the language specific learned dictionaries is chosen according to overall complexity of the system. The flow diagram of the proposed ensemble of learned-exemplar dictionary based LID approach is illustrated in Fig. 2.

5 Experiment-I

5.1 Database

The NIST 2007 language recognition evaluation (LRE) (2007) dataset contains 14 target languages in general test set condition. It consists of 7530 spoken utterances of 30, 10 and 3 s duration including the out-of-set data. In this work, the closed set task is chosen with test utterances having 30 s duration. Owing to limited training examples in German, Farsi and Tamil languages, the results are reported for 1837 test utterances of 11 languages excluding these three languages.

The development and training datasets consist of 2493 and 550 segments respectively. These datasets are collected from conversational telephone speech (CTS) in 11 languages. The development data set includes the speech data extracted from multiple corpora: OGI-multilingual, previous NIST 1996, 2003, 2005 LREs and NIST 2004, 2005, 2006, 2008 speaker recognition evaluations (SREs). The training data set contains NIST 2007 LRE supplementary training data and some data from SRE databases which are not included in the development dataset. The development and training data are separately pooled language-wise in feature space after the removal of silence segments. Among all languages in the development data, Bengali has a minimum of 73 utterances. So, the biggest size balanced exemplar dictionary is created using 70 examples per language.

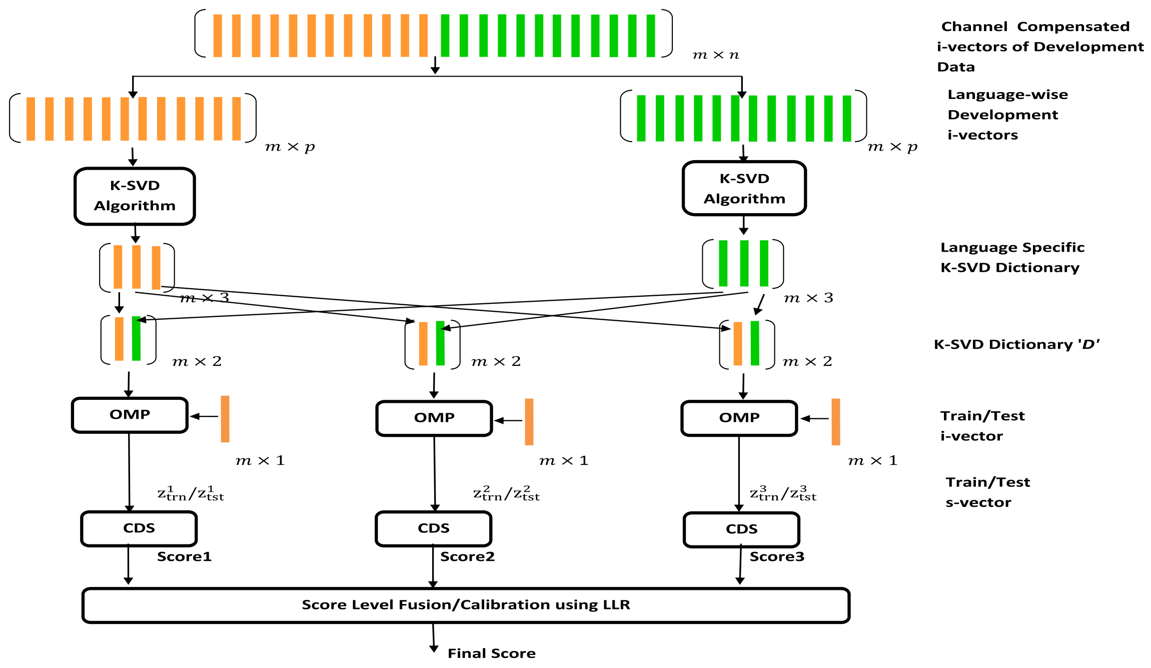

Fig. 2 Flow diagram of the proposed ensemble learned-exemplar dictionary based LID approach. For ease of illustration, the session/channel compensated i-vectors from only two language classes are shown and depicted by orange and green colors. Given the development data, first the language specific K-SVD dictionaries are created. The ensemble exemplar dictionaries are then derived by selecting single unique column from each of the language specific learned dictionaries. The remaining flow of the system is identical to that of shown in Fig. 1. (Color figure online)

5.2 MFCC feature

The speech signals are analyzed with 20 ms Hamming window with 10 ms shift and a pre-emphasis factor of 0.97. Each frame is converted into 13-dimensional (c_1–c_{13}) base mel frequency cepstral coefficients (MFCCs) (Davis and Mermelstein 1980) by considering 22 logarithmically spaced filterbank. Base MFCCs are concatenated with their first- and second-order derivatives to form 39-dimensional final feature vector. To reduce the channel effect, the cepstral mean and variance normalization are applied to the feature.

5.3 Language modeling and fusion of scores

A language independent GMM-UBM of 1024 Gaussians is employed to build the LID system by pooling approximately 1 h of data from all 11 languages. A total variability matrix of rank 400 is trained on development data and is used for computation of the i-vectors as the utterance representation. The dimension of i-vector is selected to be 400. The CDS classifier based score for the ensemble dictionaries are fused (or calibrate) using multi-class logistic regression employing the FoCal toolkit (Brummer 2007) prior to final decision.

5.4 Parameter tuning

In the proposed LID systems, the parameters like the number of examples involved in creating the dictionaries and the sparsity used in dictionary learning as well as the coding of the targets are tuned. The tuning profiles for single dictionary cases are shown in Figs. 3 and 6, while the same for the ensemble cases are shown in Figs. 4 and 7.

5.5 Results and discussion

The developed LID systems are evaluated using the average detection cost function $\boldsymbol{C}_{\text{avg}}$ as defined in the NIST 2007 (The 2007 NIST Language Recognition Evaluation Plan 2007). Separate systems are developed on the i-vector and the JFA latent vector as the low dimensional representation of the spoken language. In the following, we present the performance evaluation of the proposed ensemble exemplar and ensemble learned-exemplar dictionaries based LID systems.

5.5.1 Ensemble exemplar and learned dictionary based LID systems on the i-vectors

For setting up the proposed single dictionary based LID approaches, one is faced with following questions: (a) how much data be sufficient for creating the exemplar dictionary

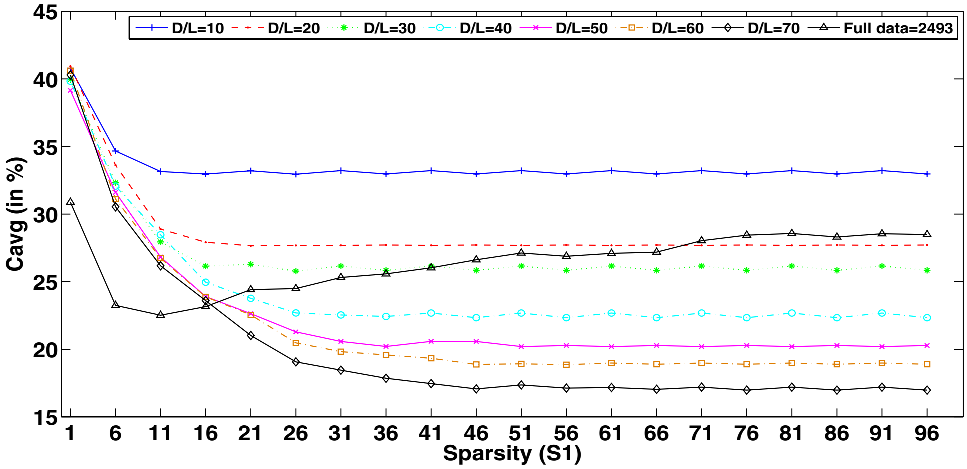

Fig. 3 Detection performances of single exemplar dictionary based LID system on the i-vector representation for varying dictionary sizes and the sparsity used in coding (*D/L* data used per language, *S1* sparsity value used in coding of the session/channel compensated targets; number of languages = 11; full data = 2493 examples from all 11 languages)

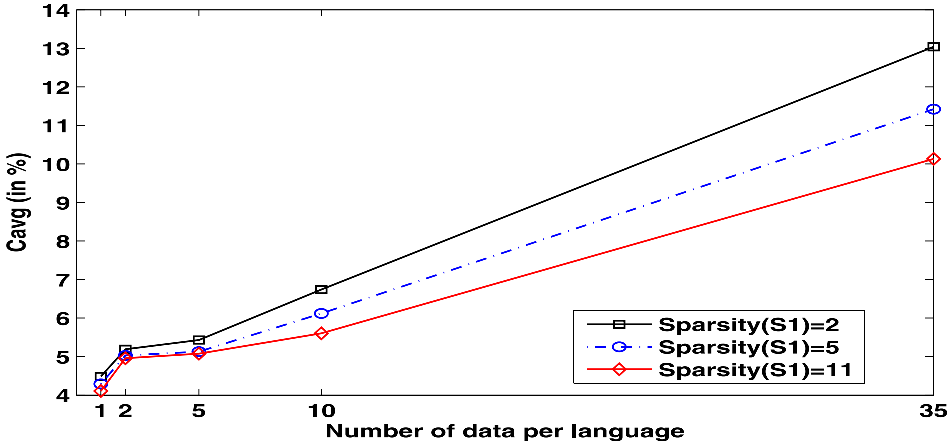

Fig. 4 The effect of dictionary size in ensemble exemplar dictionary based LID system on the i-vector representation. The equal number of examples per language is considered, thus the size of the component dictionary is equal to the number of data per language times the number of languages

with good generalizations, (b) what if the dictionary has unbalanced data across the languages, and (c) what would be the appropriate value of the sparsity for the target sparse coding and the dictionary learning. A study was made in these regards and the resulting performances in term of the average detection cost are shown in Fig. 3. It can be inferred that a dictionary created using larger and balanced data leads to better detection performance.

Figure 4 shows the effect of the size of exemplar dictionary (number of data per language × number of languages) in the ensemble dictionary based LID approach. The multiple balanced data sub-dictionaries are created out of 70 developmental examples per language. The best performance for the proposed ensemble based LID system is obtained when the sub-dictionary size is kept as 11 (i.e., having 1 example per language).

The effect of number of exemplar dictionaries in the ensemble based LID system can be noted from Fig. 5. With each dictionary being of size 11, as expected the best performance is achieved for full diversity case (i.e., number of dictionaries being 70). The sparsity in coding of targets (S1) is also tuned and three values (1, 5 and 11) are reported. It is observed that the detection performances for S1 = 5 turn out to be quite close to those for S1 = 11 (no sparsity case) while showing improvement over those for S1 = 1 (SRC case).

Figure 6 shows the detection performances of the single learned-exemplar dictionary for varying size dictionaries being created using full (having unequal examples per language) and balanced (70 examples per language) data. It is to note that, similar to the simple exemplar dictionary case, the best performances are achieved when balanced data is used in learning language specific dictionaries and the size of the resulting learned-exemplar dictionary is kept as 11 (i.e., when single column K-SVD learned dictionary is created for each of the target languages).

The ensemble exemplar dictionary based LID system is noted to outperform not only the simple exemplar dictionary based system but also the the single learned-exemplar dictionary based system. But the idea of creating multiple sub-dictionaries can also be extended to learned-exemplar dictionary case. Figure 7 shows the performance comparison between ensemble exemplar and ensemble learned dictionary based LID systems for varying number of dictionaries. It can be observed that for the same number of dictionaries, the

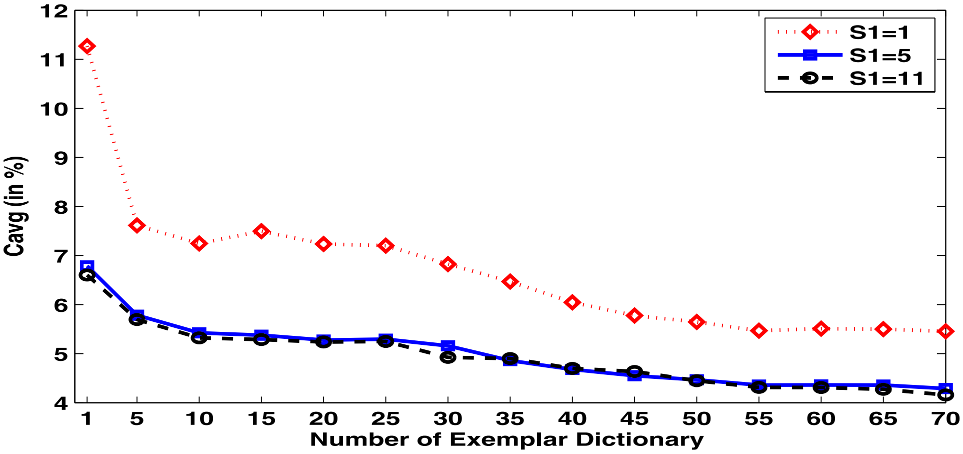

Fig. 5 The effect of number of exemplar dictionary in multiple exemplar based LID system on the i-vector. Each dictionary is of size 11 (i.e., one example per language). *S1* sparsity in coding of the targets

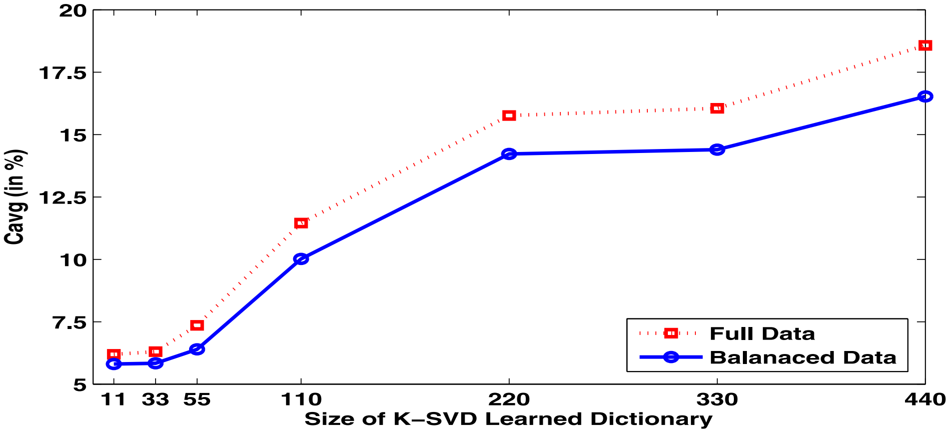

Fig. 6 Detection performances of single learned exemplar dictionary based LID system on the i-vector representation for varying size dictionaries created using full (unbalanced) data: unequal examples per language and balanced data: 70 examples per language (dict. learning sparsity = 5; dict. learning iterations = 50; target coding sparsity = 5)

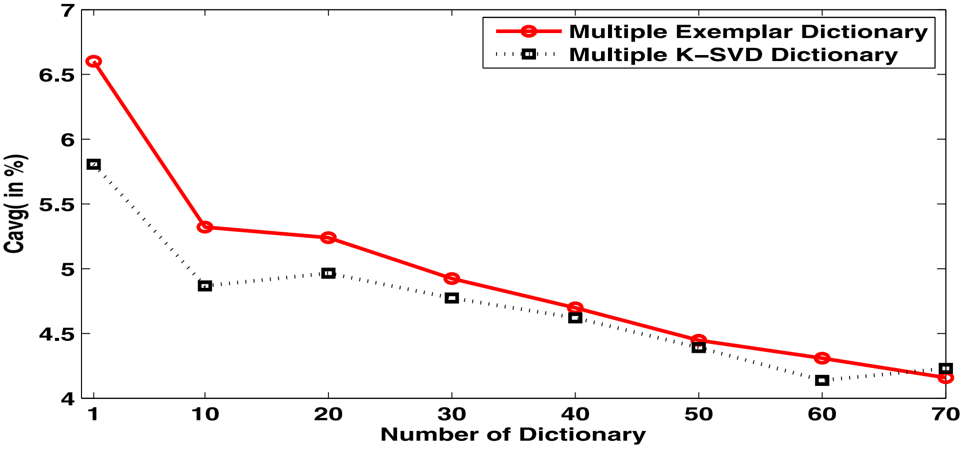

Fig. 7 Performance comparison between ensemble simple-exemplar/learned-exemplar dictionary based LID systems on the i-vector representation for varying number of dictionaries (dict. size = 11; dict. learning iterations = 50; dict. learning sparsity = 5; Target coding sparsity = 11)

learned-exemplar based LID system outperforms the simple exemplar based LID system with difference between performances diminishing with increasing number of dictionaries.

5.5.2 Ensemble exemplar and learned dictionary based LID systems on the JFA latent vector

In JFA, the language and session/channel variabilities are modeled separately whereas in the i-vector approach both are captured simultaneously. Therefore one need to apply session/channel compensation to the i-vectors before creating the dictionaries. On the other hand, the JFA latent vector denoting the language space can be directly used for creating the dictionaries. In order to avoid any leftover nuisance in JFA latent vector, it is preferable to apply suitable session/channel compensation on JFA latent vector too. As the LID task comprises of 11 languages, the size of latent vector in the JFA is also taken as 11. On employing these small size representations in place of 400-dimensional i-vectors, the computational cost of the proposed LID systems can be reduced hugely. This motivated us to explore the proposed ensemble exemplar and ensemble learned-exemplar dictionaries based LID approaches on the JFA latent vector as the utterance representation.

Figure 8 shows the detection performance of ensemble exemplar dictionary based LID system on the JFA latent vector for varying number of dictionaries created. The results are reported for the balanced data dictionaries and three target coding sparsity (S1) values (1, 5 and 11). The best performance is obtained in the case of 70 dictionaries (i.e., each dictionary having 1 atom per language) and S1 = 11. A similar trend has also been noted in case of the i-vectors as the utterance representation.

The detection performances of single learned-exemplar dictionary on the JFA latent vector for varying dictionary sizes and two sparsity values in coding of the targets are shown in Fig. 9. It can be observed that, similar to the i-vector case, the best performance is obtained for the learned exemplar dictionary having 11 atoms only.

Figure 10 shows the performance comparison between ensemble simple-exemplar/learned-exemplar dictionary based LID systems on the JFA latent vector representation. For the same number of dictionaries, the learned-exemplar based LID system outperforms the simple exemplar based LID system with difference between performances diminishing with increasing number of dictionaries. This trend is quite similar to that noted in case of the i-vector based representations as shown in Fig. 7.

Table 1 summaries the results for the proposed single and ensemble exemplar/learned-exemplar dictionary based LID systems. For contrast purpose, the performances of the i-vector based LID systems using CDS and SRC classifiers are also reported. The proposed ensemble exemplar/

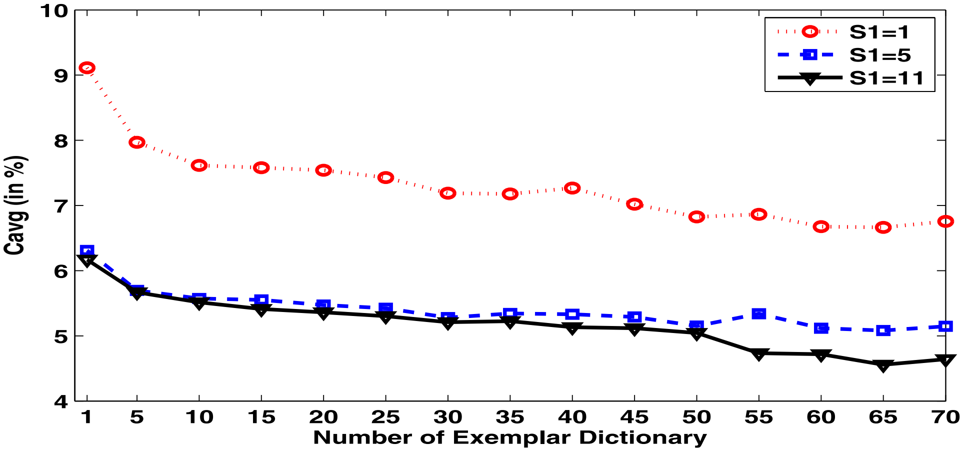

Fig. 8 The effect of number of exemplar dictionary in multiple exemplar based LID system on the JFA latent vector. Each dictionary is of size 11 and contains one example per language. *S1* sparsity used in coding of the targets

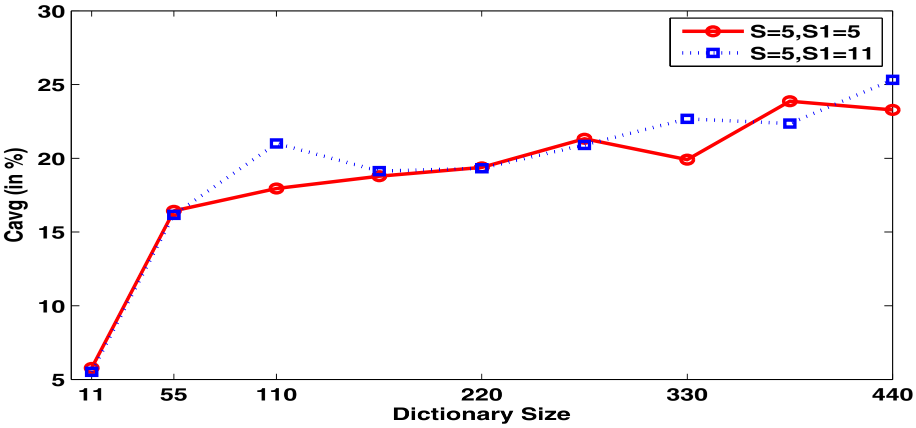

Fig. 9 Detection performances of single learned-exemplar dictionary based LID system on the JFA latent vector representation for varying size dictionaries created using balanced data (*S* dictionary learning sparsity; *S1* target coding sparsity; number of iterations in dictionary learning = 50)

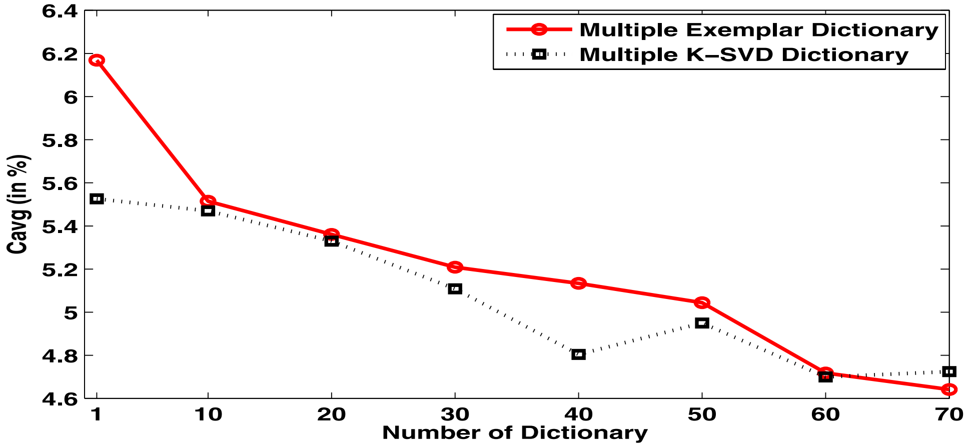

Fig. 10 Performance comparison between ensemble simple-exemplar/learned-exemplar dictionary based LID systems on the JFA latent vector for varying number of dictionaries (dict. size = 11; dict. learning iterations = 50; dict. learning sparsity = 5; target coding sparsity = 11)

Table 1 Performance comparison between session/channel compensated i-vector/JFA latent vector based, single exemplar/learned-exemplar dictionary based and ensemble of simple-exemplar/learned-exemplar dictionary based LID systems

LR system	Utterance	Classifier representation	Type /number/size dictionary	$C_{avg}(\%)$					
				N= 1		N= 60		N= 70	
				S1= 5	S1= 11	S1= 5	S1= 11	S1= 5	S1= 11
Contrast	i-Vector	SRC	Exmplr/Sngl/11	6.49	6.02	–	–	–	——
		CDS	–	**5.24**		–	–	–	–
Proposed-I	Sparse coded i-vector	CDS	Lrnd/Sngl/11	6.12	5.80	–	–	–	–
			Exmplr/Ensmbl/11	–	–	4.36	4.30	4.29	**4.15**
			Lrnd/Ensmbl/11	–	–	4.27	**4.13**	4.38	4.23
Proposed-II	Sparse coded JFA latent vector	CDS	Lrnd/Sngl/11	5.78	5.52	–	–	–	–
			Exmplr/Ensmbl/11	–	–	5.12	4.71	5.15	**4.64**
			Lrnd/Ensmbl/11	–	–	5.42	**4.70**	5.23	4.72

Score level fusion/caliration performed using multi-class logistic regression (parameters: dict. learning sparsity S = 5; dict. learning iterations = 50; target coding sparsity S1 = 5/11; number of dictionaries N = 1/60/70)

Good results obtained for existing and the proposed LID sytems are given in bold

learned-exemplar dictionary based LID systems are found to be more effective than the single exemplar/learned-exemplar dictionary based ones. These improvements are attributed to the diversity in the CDS scores for the given target obtained over multiple dictionaries. On comparing with the i-vector with CDS classifier approach, the ensemble learned-exemplar dictionary based LID systems have resulted in a relative improvement of 21.18% in C_{avg}. In case of single K-SVD learned dictionary based LID system, a relative improvement of 77.60% in C_{avg} is observed comparing with single exemplar dictionary based system. The atoms in the K-SVD dictionary are iteratively learned to better fit for the data, alternating between sparse coding and dictionary update stages. In this way, some of nuisances in the data are removed. This is the reason why the LID systems based on learned-exemplar dictionary outperforms compared to simple-exemplar dictionary. However, both ensemble exemplar and learned-exemplar dictionary based LID systems yield the similar performances. With increasing partitions of the development set, the data in each of the subsets gets reduced. The dictionary learning with reduced data is not effective, and hence the noted improvements are caused due to diversity only.

6 Experiment-II

6.1 Database

Here, the NIST 2007 language recognition evaluation (LRE) (2007) dataset is used considering 12 languages excluding German and Tamil languages. In this, the closed set task is chosen with test utterances having 30 s duration. The results are reported for 1917 test utterances of 12 languages. The training dataset include 4529 utterances collected from conversational telephone speech (CTS) in 12 languages. The training data set includes the speech data extracted from multiple corpora: OGI-multilingual, previous NIST 1996, 2003, 2005 LREs and NIST 2004, 2005, 2006, 2008 speaker recognition evaluations (SREs). The training data set contains NIST 2007 LRE supplementary training data and some data from SRE databases.

6.2 Shifted delta cepstral feature

Speech signal is processed in frames of 20ms duration at 10ms frame rate. For each frame, standard 7 Mel frequency cepstral coefficients (MFCCs) (Davis and Mermelstein 1980) are computed using 22 logarithmically spaced filter banks. The rasta filtering (Hermansky and Morgan 1994) followed by cepstral mean and variance normalization (CMVN) (Viikki and Laurila 1998) is applied on MFCCs after silence removal. Then, 49 dimensional shifted delta cepstral (SDC) (Torres-Carrasquillo et al. 2002) coefficients using 7-1-3-7 scheme are obtained and concatenated to seven static MFCCs to form 56 dimensional feature vector.

6.3 Language modeling, classifiers and dictionary learning

The language modeling steps are similar to those discussed for experiment-I. The various classifiers are investigated on the i-vector utterance representation. In ensemble of learned-exemplar dictionary based LID approach, online dictionary learning employing ENet based SR is used. The SRC classifier for each of the parallel unit is applied unlike CDS used in experiment-I. Here, we have used least angle regression (LARS) to solve the ENet based sparse coding problem. The SRC based scores are calibrated using Gaussian backend followed by multi-class logistic regression. The FoCal toolkit (Brummer 2007) is used for this purpose. The regularization parameters λ_1 and λ_2 are chosen to be 0.01 and 0.2 respectively.

6.4 Results

Table 2 summarizes the results for channel compensated i-vector utterance representation with various classifiers. The SRC over an exemplar dictionary, learned-exemplar dictionary employing for single and ensemble dictionary based LID are compared.

7 Experiment-III

In this, we have used the similar steps as discussed for experiment II. Only difference lies in the database used. We consider only 12 languages which are sub-set of the NIST-2009 Language Recognition Evaluation (LRE) (2009) corpora due to limited training data in remaining languages. The sub-set includes American English, Cantonese Chinese, Mandarin and Russian, Farsi, Hindi, Korean, Vietnamese, Creole, Georgian, Turki, and Spanish languages. The training data is collected from the conversational telephone speech (CTS) only, which includes NIST SRE (2004, 2005, 2006 and 2008), NIST LRE 2007 supplementary and Babel (Roach et al. 1996) data-sets. The training dataset contains only 250 speech utterances from each of the language. The test data includes both VoA and CTS data from these 12 languages. The experiments are performed in closed set condition on 3, 10 and 30 s duration segments.

7.1 Results

The performance of the proposed ensemble of exemplar and learned-exemplar dictionary employing SRC for LID task are summarized in Table 3. The three performance measure namely average detection cost function $\boldsymbol{C}_{\text{avg}}$, multi-class $\boldsymbol{C}_{\text{llr}}$ and identification rate (IDR) are considered. Both i-vector

Table 2 Performance comparison between standard WCCN compensated i-vector with different classifiers

LR system	Utterance representation	Dictionary type/number/size	Classifier	$\%\boldsymbol{C}_{\text{avg}}$	mc- $\boldsymbol{C}_{\text{llr}}$ (in bits)
Contrast	i-Vector	–	SVM	4.20	0.57
			MLR	3.42	0.47
			GG	3.39	0.46
			CDS	3.24	0.45
Propose	i-Vector	Exmplr/sngl/4529	SRC	3.56	0.48
		Lrnd/sngl/24	SRC	3.21	0.45
		Lrnd/Ensmbl/24	SRC	**3.05**	**0.44**

(*SVM* support vector machine, *MLR* multi-class logistic regression, *GG* generative Gaussian,*CDS* cosine distance scoring, *SRC* sparse representation based classification), the single learned-exemplar dictionary based LID and ensemble of learned-exemplar dictionary based LID

The 2007 NIST evaluation data-set in closed set condition on 30 s segments are used. The scores are calibrated using Gaussian backed followed by MLR

Bold values show the best performances obtained for proposed system

Table 3 Performance comparison between standard WCCN compensated i-vector with different classifiers

LR system	Representation utterance	Dictionary type/number/size	Classifier	**30** s			**10** s			**3** s		
				%$\boldsymbol{C}_{\text{avg}}$	mc- $\boldsymbol{C}_{\text{llr}}$	IDR (%)	%$\boldsymbol{C}_{\text{avg}}$	mc-$\boldsymbol{C}_{\text{llr}}$	IDR (%)	%$\boldsymbol{C}_{\text{avg}}$	mc-$\boldsymbol{C}_{\text{llr}}$	IDR (%)
Contrast	i-vector		SVM	5.30	0.69	83.65	12.71	1.53	64.44	25.89	2.58	37.83
			G-PLDA	4.69	0.61	85.54	11.78	1.42	66.75	24.66	2.58	39.86
		–	MLR	4.52	0.59	86.00	11.57	1.38	67.29	24.24	2.57	39.85
			GG	4.50	0.58	86.41	11.35	1.37	67.65	24.30	2.56	40.15
			CDS	4.48	0.57	86.52	11.43	1.36	67.80	24.02	2.55	40.47
		Exmplr/sngl/3000	SRC	4.55	0.60	86.27	11.78	1.41	67.22	24.84	2.60	39.84
Proposed	i-vector	Exmplr/Ensmbl(125)/24	SRC	3.68	0.47	89.73	10.24	1.26	71.00	22.72	2.45	42.77
		Lrnd/sngl/24	SRC	4.44	0.57	86.47	11.45	1.37	67.57	24.10	2.55	40.58
		Lrnd/Ensmbl(100)/24	SRC	**3.01**	**0.39**	**92.11**	**9.25**	**1.14**	**74.82**	**21.45**	**2.34**	**47.31**
	JFA latent vector	Exmplr/sngl/3000	SRC	10.75	1.33	74.20	18.16	2.12	56.56	31.08	3.03	33.08
		Exmplr/Ensmbl(125)/24	SRC	5.80	0.74	82.07	13.68	1.63	62.74	27.15	2.77	36.32
		Lrnd/sngl/24	SRC	6.62	0.85	80.52	14.48	1.72	60.95	28.06	2.83	35.36
		Lrnd/Ensmbl(100)/24	SRC	**5.72**	**0.74**	**82.27**	**13.62**	**1.63**	**62.13**	**27.41**	**2.77**	**35.86**

(*SVM* support vector machine, *G-PLDA* Gaussian PLDA, *MLR* multi-class logistic regression, *GG* generative Gaussian, *CDS* cosine distance scoring, *SRC* sparse representation based classification), the single learned-exemplar dictionary based LID and ensemble of exemplar and learned-exemplar dictionary based LID

The 2009 NIST evaluation data-set in closed set condition on 3, 10 and 30 s segments are used. The scores are calibrated using Gaussian backed followed by MLR. The i-vector dimension is 400 while JFA latent vector size is 12

Bold values show the best performances obtained for proposed system

Table 4 Performance comparison between PTN and the proposed ensemble of exemplar and learned-exemplar dictionary employing i-vector based LID systems with three test conditions according to length (dev-all: full length utterance, dev-1 and dev-3 s are extracted from full length utterance)

LR	Dev-all sec			Dev-3 sec			Dev-1 sec		
System	%C_{avg}	mc- C_{llr}	IDR (%)	%C_{avg}	mc-C_{llr}	IDR (%)	%C_{avg}	mc-C_{llr}	IDR (%)
PTN	1.78	0.23	94.58	3.21	0.39	90.58	**8.41**	**0.96**	**77.00**
ivector+CDS	3.08	0.39	90.80	4.74	0.57	86.45	12.19	1.35	67.74
Exmplr/Ensmbl(200)/20	3.10	0.37	91.10	4.83	0.58	86.28	12.97	1.42	67.12
Exmplr/Ensmbl(500)/20	**1.44**	**0.18**	**95.52**	**2.88**	**0.35**	**91.60**	10.59	1.19	72.53
Lrnd/Ensmbl(100)/20	1.65	0.20	95.44	2.95	0.36	91.74	10.31	1.18	73.92
Lrnd/Ensmbl(200)/20	1.12	0.14	96.86	2.17	0.27	93.72	9.08	1.05	76.72
Lrnd/Ensmbl(300)/20	**0.88**	**0.11**	**97.28**	**1.84**	**0.23**	**94.51**	8.58	0.99	77.88

The scores are calibrated using Gaussian backed followed by MLR

and JFA latent vectors are used as the utterance representation. For contrast purpose, various non-sparse classifiers are investigated on channel compensated i-vector. The SRC over an exemplar dictionary, learned-exemplar dictionary employing single dictionary based LID are also compared. The i-vector based LID system employing CDS classifier is found to be better than SVM, G-PLDA, MLR, GG and SRC over single exemplar dictionary (created by concatenating 3000 training data from 12 languages) based LID system. However, there is a slight degradation in the CDS based LID system on comparing with single learned dictionary consisting of only 24 dictionary atoms (two dict. atoms per language). It is also observed that ensemble of exemplar dictionary based approach outperforms a single exemplar dictionary. The multiple exemplar dictionaries are created by partitioning whole training data (i.e.,3000) into 125 exemplar dictionaries, each with 24 training data (two data per language). Thus, improvement achieved with the ensemble of exemplar dictionary approach is attributed to the diversity achieved in sparse coding of target over multiple dictionary. This approach may not be feasible if the size of the training data is large, because it requires many sparse coding of the test feature vector. To address this problem, we further explore ensemble of learned-exemplar dictionary and found to outperforms ensemble of exemplar dictionary with only 100 numbers of dictionary. The multiple small sized learned-exemplar dictionaries are created by concatenating equal number of learned dictionary atoms from each of the language specific learned dictionary. Thus, the number of small sized learned-exemplar dictionary is restricted by the size of language specific dictionary.

8 Experiment-IV

The multi-lingual speech corpus alongside its transcription for oriental languages are collected from AP17-OL3 and AP16-OL7 as per AP17-OLR Challenge (Tang et al. 2017). The AP17-OL3 includes three languages Kazakh, Tibetan and Uyghur while AP16-OL7 includes seven languages Cantonese, Mandarin, Indonesian, Japanese, Russian, Korean and Vietnamese. Thus, a total of 10 languages in closed set condition with three test duration of 1s, 3s and full length utterances are considered. The details of the data distribution can be found in Tang et al. (2017). The performance of the proposed ensemble of exemplar and learned-exemplar dictionary employing i-vector based LID systems are compared with phonetic temporal neural (PTN) based LID system. The THCHS30 Chinese speech database (Wang and Zhang 2015) is freely available database and used to train the TDNN phonetic model. The 20 dimensional MFCCs including log energy are augmented with first and second order derivatives, resulting in 60-dimensional feature vectors. The UBM contains 2048 Gaussian components and i-vectors of 400 dimension are extracted. The phonetic DNN is a six hidden layer TDNN structure with p-norm activation function, and the LSTM-RNN based LID model. The raw feature of PTN based LID is 40-dimensional Fbanks, with a symmetric 4-frame window for the TDNN and a symmetric 2-frame window for the LSTM-RNN to splice neighboring frames. Each TDNN layer has 2048 units. The number of cells of the LSTM is taken to be 1024. The kaldi toolkit is used for the development of i-vector and PTN based LID systems. The scores are calibrated using Gaussian backed followed by MLR.

8.1 Results

Table 4 summarizes the results for PTN and the proposed ensemble of exemplar and learned-exemplar dictionary employing i-vector based LID systems considering ten languages. The results are reported in closed set condition with three test durations (dev-all: full length utterance, dev-1 and dev-3 s are extracted from full length utterance). On comparing ensemble of exemplar dictionaries having 200 and 500 dictionaries, the system having 500 dictionaries shows the improved performance. The ensemble of exemplar dictionaries having 500 exemplar dictionaries outperforms PTN based LID system on dev-all and dev-3 sec. The performance is further improved with ensemble of learned-exemplar dictionary based approach. It is noted that the number of learned-exemplar dictionaries required is comparatively less than that of exemplar dictionaries in ensemble approach. Also, the performance with 300 learned-exemplar dictionaries on 1 s is comparable to PTN based LID system.

9 Conclusion

In this work, we have explored the single and multiple exemplar/learned dictionaries based sparse representation on the i-vector for LID task. The proposed ensemble exemplar and learned-exemplar dictionary based LID systems have been found to be effective than the ones based on single exemplar/learned dictionary. The single learned dictionary based LID has been noted to outperform the single exemplar based LID. The performances of both ensembles of simple-exemplar and learned-exemplar dictionaries are found to be similar in limited training data scenario. However, ensembles of learned-exemplar dictionaries outperforms the ensembles of simple-exemplar dictionaries with large training data. Thus, improvement achieved with the ensemble of dictionary approach is attributed to the diversity achieved in sparse coding of target over multiple dictionary. Furthermore, the effectiveness of the proposed algorithms is investigated on low dimensional JFA latent vector, and similar performance trends are observed in comparing with i-vector based LID systems.

Acknowledgements The authors would like to thank the linguistic data consortium (LDC) for providing the LDC2014S06 2009 NIST LRE test set under LDC Spring 2016 data scholarships program.

References

Aharon, M., Elad, M., & Bruckstein, A. (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, *54*(11), 4311–4322.

Ambikairajah, E., Li, H., Wang, L., Yin, B., & Sethu, V. (2011). Language identification: A tutorial. *IEEE Circuits and Systems Magazine*, *11*(2), 82–108.

Brummer, N. (2007). Focal multi-class: Toolkit for evaluation, fusion and calibration of multi-class recognition scores—tutorial and user manual. http://sites.google.com/site/nikobrummer/focalmulticlass.

Davis, S., & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *28*(4), 357–366.

Dehak, N., Torres-Carrasquillo, P. A., Reynolds, D., & Dehak, R. (2011). Language recognition via i-vectors and dimensionality reduction. In: *Proceedings of Interspeech* (pp. 857–860).

Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., & Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech and Language*, *19*(4), 788–798.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological), 39*, 1–38.

Haris B.C., & Sinha, R. (2012) Speaker verification using sparse representation over ksvd learned dictionary. In: *Proceedings of National Conference on Communications (NCC)* (pp. 1–5).

Haris, B. C., & Sinha, R. (2015). Robust speaker verification with joint sparse coding over learned dictionaries. *IEEE Transactions on Information Forensics and Security*, *10*(10), 2143–2157.

Hatch, A.O., Kajarekar, S., & Stolcke, A. (2006). Within-class covariance normalization for svm-based speaker recognition. In: *Proceedings of the ICSLP* (pp. 1471–1474).

Hermansky, H., & Morgan, N. (1994). Rasta processing of speech. *IEEE Transactions on Speech and Audio Processing*, *2*(4), 578–589.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine*, *20*(8), 832–844.

Jiang, B., Song, Y., Guo, W., & Dai, L.R. (2012). Exemplar-based sparse representation for language recognition on i-vectors. In: *Proceedings of ISCA Interspeech.*

Kenny, P. (2010). Bayesian speaker verification with heavy-tailed priors. In: Odyssey, p. 14.

Kenny, P., Boulianne, G., & Dumouchel, P. (2005). Eigenvoice modeling with sparse training data. *IEEE Transactions on Speech and Audio Processing*, *13*(3), 345–354.

Kenny, P., Boulianne, G., Ouellet, P., & Dumouchel, P. (2007). Joint factor analysis versus eigenchannels in speaker recognition. *IEEE Transactions on Audio Speech and Language Processing*, *15*(4), 1435–1447.

Kua, J., Ambikairajah, E., Epps, J., & Togneri, R. (2011). Speaker verification using sparse representation classification. In: *Proceedings of IEEE ICASSP* (pp. 4548–4551).

Mairal, J., Bach, F., Ponce, J., & Sapiro, G. (2010). Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, *11*(Jan), 19–60.

Martinez, D., Plchot, O., Burget, L., Glembek, O., & Matejka, P. (2011). Language recognition in i-vectors space. In: *Proceedings of Interspeech* (pp. 861–864).

Naseem, I., Togneri, R., & Bennamoun, M. (2010). Sparse representation for speaker identification. In: *Proceedings of the 20th International Conference on Pattern Recognition (ICPR)* (pp. 4460–4463).

Ng, A.Y. (2004). Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In: *Proceedings of the Twenty-first International Conference on Machine Learning*, p. 78. ACM.

Pati, Y.C., Rezaiifar, R., & Krishnaprasad, P.S. (1993). Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In: *Proceedings of 27th Asilomar Conference Signals, Systems and Computers*, Pacific Grove, CA (vol. 1, pp. 40–44). https://doi.org/10.1109/ACSSC.1993.342465.

Prince, S.J., & Elder, J.H. (2007). Probabilistic linear discriminant analysis for inferences about identity. In: *IEEE 11th International Conference on Computer Vision*, 2007. ICCV 2007 (pp. 1–8). IEEE.

Roach, P., Arnfield, S., Barry, W., Baltova, J., Boldea, M., Fourcin, A., Gonet, W., Gubrynowicz, R., Hallum, E., Lamel, L., et al. (1996). Babel: An eastern european multi-language database. In: *Proceedings of Fourth International Conference On Spoken Language*, 1996. ICSLP 96 (vol. 3, pp. 1892–1893). IEEE.

Singh, O.P., Haris B.C., & Sinha, R. (2013). Language identification using sparse representation: A comparison between gmm supervector and i-vector based approaches. In: *Proceedings of Annual IEEE India Conference (INDICON)* (pp. 1–4).

Singh, O.P., & Sinha, R. (2017). Sparse representation classification based language recognition using elastic net. In: *2017 4th*

International Conference on Signal Processing and Integrated Networks (SPIN).

Tang, Z., Wang, D., Chen, Y., & Chen, Q. (2017). Ap17-olr challenge: Data, plan, and baseline. CoRR arXiv:abs/1706.09742.

Tang, Z., Wang, D., Chen, Y., Li, L., & Abel, A. (2017). Phonetic temporal neural model for language identification. arXiv preprint arXiv:1705.03151.

The 2007 NIST Language Recognition Evaluation Plan. (2007). http://www.itl.nist.gov/iad/mig//tests/lre/2007/LRE07EvalPlan-v8b.pdf. Accessed 28 Feb 2015.

The 2009 NIST Language Recognition Evaluation Plan. (2009). http://www.itl.nist.gov/iad/mig//tests/lre/2009/LRE09EvalPlan-v6.pdf. Accessed 5 July 2016.

Torres-Carrasquillo, P.A., Singer, E., Kohler, M.A., Greene, R.J., Reynolds, D.A., & Deller, Jr., J.R. (2002). Approaches to language identification using gaussian mixture models and shifted delta cepstal features. In: *Proceedings of ICSLP*.

Vapnik, V. (2013). *The nature of statistical learning theory*. Berlin: Springer.

Viikki, O., & Laurila, K. (1998). Cepstral domain segmental feature vector normalization for noise robust speech recognition. *Speech Communication*, *25*(1), 133–147.

Wang, D., & Zhang, X. (2015). Thchs-30 : A free chinese speech corpus. http://arxiv.org/abs/1512.01882.

Wright, J., Yang, A., Ganesh, A., Sastry, S., & Ma, Y. (2009). Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(2), 210–227.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320.