Foreign Accent Detection from Spoken Finnish Using i-Vectors

Hamid Behravan, Ville Hautamäki and Tomi Kinnunen

School of Computing, University of Eastern Finland, Joensuu, Finland
{behravan, villeh, tkinnu}@cs.uef.fi

Abstract

I-vector based recognition is a well-established technique in state-of-the-art speaker and language recognition but its use in dialect and accent classification has received less attention. We represent an experimental study of i-vector based dialect classification, with a special focus on foreign accent detection from spoken Finnish. Using the CallFriend corpus, we first study how recognition accuracy is affected by the choices of various i-vector system parameters, such as the number of Gaussians, i-vector dimensionality and reduction method. We then apply the same methods on the Finnish national foreign language certificate (FSD) corpus and compare the results to traditional Gaussian mixture model - universal background model (GMM-UBM) recognizer. The results, in terms of equal error rate, indicate that i-vectors outperform GMM-UBM as one expects. We also notice that in foreign accent detection, 7 out of 9 accents were more accurately detected by Gaussian scoring than by cosine scoring.

Index Terms: Dialect recognition, foreign accent recognition, i-vector, GMM-UBM, Finnish language

1. Introduction

A spoken language considerably varies in terms of its regional dialects and accents. *Dialect* refers to linguistic variations of a language, while *accent* refers to different ways of pronouncing a language within a community [1]. Accurate recognition of dialect or accent prior to automatic speech and language recognition may help in improving recognition accuracy by speaker and language model adaptation [2, 3]. Furthermore, in modern services based on user-agent voice commands, connecting a user to the agents with similar dialect or accent will produce a more user-friendly environment [2]. In the context of immigration screening, it may be helpful to verify semi-automatically whether an applicant's accent corresponds to accents spoken in a region he claims he is from. There is a clear need for accurate, automatic characterization of spoken dialects and accents.

Typical dialect and accent recognizers use either *acoustic* or *phonotactic* modeling. In the former approach, acoustic features such as *shifted delta cepstra* (SDC), are used with bag-of-frames models such as *universal background model* (UBM) with adaptation [4, 5]. The latter approach is based on the hypothesis that dialects or accents differ in terms of their phone sequence distributions. It uses phone recognizer outputs, such as N-gram statistics, together with language modeling backend [6, 7]. We focus on the acoustic approach for reasons of simplicity and computational efficiency.

Among the multitude of choices for acoustic modeling, *i-vector* approach [8] has proven successful in both speaker and language recognition [9, 10, 11]. It is rooted on Bayesian *factor analysis* technique which forms a low-dimensional *total variability space* containing both speaker and channel variabilities. To tackle inter-session and inter-channel variability, i-vector approach is usually combined with techniques such as *within-class covariance normalisation* (WCCN) [9].

Caused by more subtle linguistic variations, dialect and accent recognition are generally more difficult than language recognition [3]. Thus, it is not obvious how well i-vectors will perform on these tasks. In [12], an initial attempt to use i-vectors for accent classification using an iterative classification framework was investigated. Their results showed 68 % overall classification accuracy in fourteen British accents. In another fresh study [13], the authors compared three accent modelling approaches involving English utterances of speakers from seven different native languages. The i-vector accuracy was found comparable to sparse representation classifier (SRC) but outperformed the two other approaches.

From these preliminary studies, it appears that i-vector approach works reasonably well for English dialect and accent recognition corpus. This can be partly attributed to availability of massive development corpora including thousands of hours of spoken English utterances to train all the system hyperparameters. The present study presents a case when such resources are *not* available. It is part of an ongoing project involving foreign accent detection from spoken Finnish. To study this case, we conduct two separate experiments one for dialect and the other for foreign accent detection tasks. We first optimize the main control parameters such as the number of UBM components and i-vector dimensionality using a corpus with sufficient amount of data. We are also curious to replace the linear discriminant analysis (LDA) – used for i-vector dimensionality reduction – with *heteroscedastic* LDA, which unlike conventional LDA, takes into account the covariance matrices are not common across dialect or accent models. This enables us to reduce the i-vector dimensionality to desired values [14]. Figure 1 demonstrates the block diagram of the dialect and accent recognition system used in this work. The optimized system components are then applied to the Finnish foreign accent detection task.

2. System description

2.1. i-vector approach

i-vector modeling is inspired by the success of *joint factor analysis* (JFA) in speaker verification [8], where speaker and channel effects were modeled separately using eigenvoice (speaker subspace) and eigenchannel (channel subspace) model. But in [8], it was found that these subspaces are not completely independent, therefore a combined total variability space was introduced [15].

In the i-vector approach, the Gaussian mixture model (GMM) supervector ($\mathbf{M}$) for each dialect utterance is represented as,

$$\mathbf{M} = \mathbf{m} + \mathbf{T}\mathbf{w}, \tag{1}$$

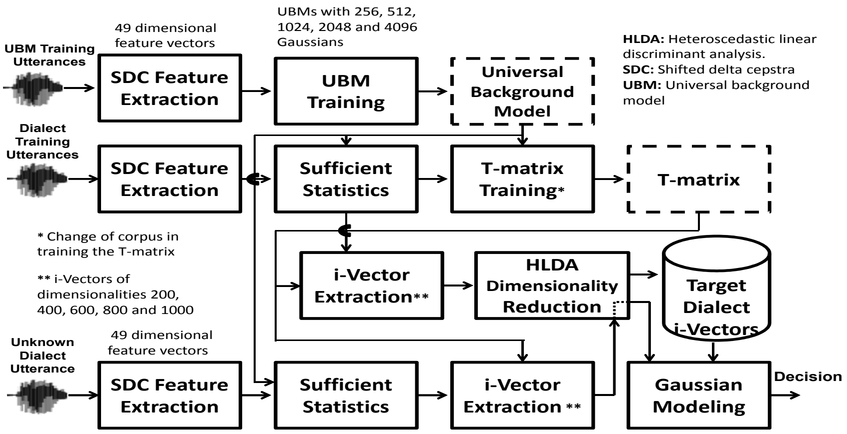
UBM Training Utterances
49 dimensional feature vectors
SDC Feature Extraction
UBMs with 256, 512, 1024, 2048 and 4096 Gaussians
UBM Training
Universal Background Model
HLDA: Heteroscedastic linear discriminant analysis.
SDC: Shifted delta cepstra
UBM: Universal background model
Dialect Training Utterances
SDC Feature Extraction
Sufficient Statistics
T-matrix Training*
T-matrix
* Change of corpus in training the T-matrix
** i-Vectors of dimensionalities 200, 400, 600, 800 and 1000
i-Vector Extraction**
HLDA Dimensionality Reduction
Target Dialect i-Vectors
Unknown Dialect Utterance
49 dimensional feature vectors
SDC Feature Extraction
Sufficient Statistics
i-Vector Extraction **
Gaussian Modeling
Decision

Figure 1: *Block diagram of i-vector dialect and accent recognition used in this work.*

where $\mathbf{m}$ is dialect- and channel-independent UBM supervector, the i-vector $\mathbf{w}$ is an independent random vector drawn from $\mathcal{N}(\mathbf{0},\mathbf{I})$, and $\mathbf{T}$ is a low-rank matrix representing the captured between-utterance variabilities in the supervector space. Because prior is normally distributed then posterior is also normal. Training the $\mathbf{T}$ matrix is similar to training the eigenvoice matrix $\mathbf{V}$ in JFA [16], except that we treat every training utterance of a given dialect model as belonging to different dialect. Extracted i-vector is then just the expectation of the posterior distribution, where $\mathbf{T}$ and $\mathbf{m}$ are the hyper-parameters.

2.2. Feature reduction with heteroscedastic linear discriminant analysis

As the extracted i-vectors contain both within- and between-dialect variation, the aim of dimensionality reduction is to project the i-vectors onto a space, where the within-dialect variation is minimal and between-dialect variation maximal. A common technique used for dimensionality reduction of i-vectors *linear discriminant analysis* (LDA), where for a L class problem, the maximum projected dimension is $L-1$. As discussed in [14], these $L-1$ dimensions do not necessarily contain all the discriminatory data for the classification task, and even if it does, it is not clear whether LDA will capture them. Furthermore, regarding our first corpus, where the recognition task is a two class problem, LDA reduces the i-vector dimension to 1, which clearly leads to incorrect results.

For these reasons, we also consider an extension of LDA, *heteroscedastic* linear discriminant analysis (HLDA) [14]. HLDA is occasionally used in speaker recognition and, unlike LDA, it deals with discriminant information presented both in the means and covariance matrices of classes. To perform dimensionality reduction, i-vector of dimension n is projected into first $p < n$ rows, $a_{j=1\ldots p}$, of $n \times n$ HLDA transformation matrix denoted by $\mathbf{A}$. The matrix $\mathbf{A}$ is estimated by an efficient row-by-row iteration [17], whereby each row is periodically re-estimated as,

$$\hat{\mathbf{a}}_k = \mathbf{c}_k \mathbf{G}^{(k)-1} \sqrt{\frac{N}{\mathbf{c}_k \mathbf{G}^{(k)-1} \mathbf{c}_k^{\mathrm{T}}}}, \tag{2}$$

where $\mathbf{c}_k$ is the k^{th} row vector of the co-factor matrix $\mathbf{C} = |\mathbf{A}|\mathbf{A}^{-1}$ for the current estimate of $\mathbf{A}$ and

$$\mathbf{G}^k = \begin{cases} \sum_{j=1}^{J} \frac{N_j}{\mathbf{a}_k \hat{\mathbf{\Sigma}}^{(j)} \mathbf{a}_k^{\mathrm{T}}} \hat{\mathbf{\Sigma}}^{(j)} & k \le p \\ \frac{N}{\mathbf{a}_k \hat{\mathbf{\Sigma}} \mathbf{a}_k^{\mathrm{T}}} \hat{\mathbf{\Sigma}} & k > p \end{cases} \tag{3}$$

where $\hat{\mathbf{\Sigma}}$ and $\hat{\mathbf{\Sigma}}^{(j)}$ are estimates of class-independent covariance matrix and covariance matrix of j^{th} model, N_j is the number of training utterances of the j^{th} model and N is the total number of training utterances. In order to avoid near-to-singular covariance matrices, principal component analysis (PCA) is applied prior to HLDA on the training i-vector features [14, 18]. The dimension of PCA is selected so that within-models scatter matrix becomes non-singular.

2.3. Cosine scoring and Gaussian scoring

We consider two scoring schemes for the inferred i-vectors. *Cosine score* [15] for two i-vectors $\mathbf{w}_{\text{test}}$ and $\mathbf{w}_{\text{target}}$ is given by their dot product $\langle \mathbf{w}_{\text{test}}, \mathbf{w}_{\text{target}} \rangle$ by the following equation, where $\mathbf{A}$ is the HLDA projection matrix, which is trained by using all the training utterances from dialects of a language:

$$\text{score}(\mathbf{w}_{\text{test}}, \mathbf{w}_{\text{target}}) = \frac{\hat{\mathbf{w}}_{\text{test}}^{\mathrm{T}} \cdot \hat{\mathbf{w}}_{\text{target}}}{\|\hat{\mathbf{w}}_{\text{test}}\| \, \|\hat{\mathbf{w}}_{\text{target}}\|}, \tag{4}$$

where $\hat{\mathbf{w}}_{\text{test}}$ is computed as

$$\hat{\mathbf{w}}_{\text{test}} = \mathbf{A}^{\mathrm{T}} \mathbf{w}_{\text{test}}. \tag{5}$$

In order to model $\hat{\mathbf{w}}_{\text{target}}$, we followed the same strategy used in [19], where $\hat{\mathbf{w}}_{\text{target}}$ is defined as

$$\hat{\mathbf{w}}_{\text{target}} = \frac{1}{N_d} \sum_{i=1}^{N_d} \hat{\mathbf{w}}_{id}, \tag{6}$$

where N_d is the number of training utterances in dialect d, and $\hat{\mathbf{w}}_i$ is the projected i-vector of training utterance i for dialect d computed the same way as in (5). In addition to cosine scoring, we also experimented with *Gaussian scoring* described in [10]. For a given i-vector $\mathbf{w}_{\text{test}}$ of a test utterance, the log-likelihood for a target dialect d is computed as,

$$ll_{\mathbf{w}_{\text{test}}} = \hat{\mathbf{w}}_{\text{test}}^{\mathrm{T}} \mathbf{\Sigma}^{-1} \mathbf{m}_d - \frac{1}{2} \mathbf{m}_d^{\mathrm{T}} \mathbf{\Sigma}^{-1} \mathbf{m}_d, \tag{7}$$

where $\mathbf{m}_d$ is the mean vector of dialect d and $\mathbf{\Sigma}$ is the common covariance matrix shared across all dialects. It is computed as,

$$\mathbf{\Sigma} = \frac{1}{D}\sum_{i=1}^{D}\frac{1}{N_d}\sum_{i=1}^{N_d}(\hat{\mathbf{w}}_{id} - \mathbf{m}_d)(\hat{\mathbf{w}}_{id} - \mathbf{m}_d)^{\mathrm{T}}, \qquad (8)$$

where

$$\mathbf{m}_d = \frac{1}{N_d}\sum_{i=1}^{N_d}\hat{\mathbf{w}}_{id}. \qquad (9)$$

and $\hat{\mathbf{w}}_{id}$ corresponds to projected i-vector of training utterance i for dialect d.

3. Experimental set-up

3.1. Corpora

CallFriend corpus [20] is a collection of unscripted conversations of 12 languages recorded over telephone lines. It includes two dialects for each target language available. All the utterances are organized into training, development and evaluation subsets. For our purposes, we selected dialects of English, Mandarin and Spanish languages and partitioned them into wave files of 30 seconds in duration, resulting in approximately 4000 splits per each subset. All the audio files have 8 KHz sample frequency.

The second corpus, *FSD* corpus [21] was developed to assess language proficiency among adults of different languages. We selected the speaking responses in Finnish. These responses correspond to 18 foreign accents. Unfortunately, as the number of utterances in some accents was not high enough to perform recognition experiments, 9 accents — Russian, Albanian, Arabic, Chinese, English, Estonian, Kurdish, Spanish, and Turkish — with enough available utterances were chosen for the experiments. The unused accents were, however, used in training UBM and the T-matrix. For our purposes, each accent set is randomly split into a test and a train set. Split was done in such a way that no speaker is placed into both test and train sets. The test set consists of (approximately) 30% of the utterances, while the training set consists of the remaining 70%. The original raw mp3 audio files were further partitioned into 30 seconds length and resampled to 8 KHz wave files.

3.2. Feature extraction

The feature extraction process consists of windowing the speech signal at 20ms length and 10ms shift filtered through Mel-scale filterbank over the band 0-4000 Hz, producing 27 log-filterbank energies. RASTA filtering is applied to log-filterbank energies and producing seven cepstral coefficients (c0-c6) via DCT. The cepstral coefficients are further normalized using cepstral mean and variance normalization (CMVN) and vocal tract length normalization (VTLN) [22], and converted into 49-dimensional shifted delta cepstra (SDC) feature vectors [23] with 7-1-3-7 parameters. Finally, non-speech frames are removed to obtain the final SDC feature vector.

3.3. GMM-UBM system

In order to have a baseline comparison with conventional dialect and accent recognition systems, we also developed a GMM-UBM system of 2048 components similar to the work presented in [24]. It consists of 10 iterations of EM and 1 iteration for adapting the UBM to each dialect model using SDC features. During the adaptation process, means, variance and weights are all updated given the training data for each dialect. In this work, UBMs are constructed per language, meaning that for each language available, UBMs are built by using all training utterances available within the dialects of a specific language. The testing procedure employs a fast scoring scheme as described in [25] to score the input utterance to each adapted dialect model.

3.4. Classifiers and evaluation metric

To investigate i-vector recognizer in dialect and foreign accent recognition tasks, we developed four testing conditions on the CallFriend corpus. The purpose of these experiments is to search for the optimal i-vector parameters for dialect recognition and, consequently, use them to report the performance of i-vector system in foreign accent recognition. For all experiments, log-likelihood scores are calibrated with multi-class logistic regression method [26] and the results are reported for both cosine scoring and Gaussian scoring classifiers.

System performance is reported in terms of *equal error rate (EER)*. It indicates the operating point at which false alarm and miss alarm rates are equal. Scores are computed by pooling out all scores from all target and non-target dialects or foreign accents. In case of FSD corpus, we also report the individual EER of each target accent.

4. Results

Table 1 lists the CallFriend performance results for selected i-vector dimensionalities. In contrast to language recognition systems [10], recognition performance improves as the i-vector dimensionality increases in both classifiers. Furthermore, the Gaussian classifier slightly outperforms cosine scoring. Our results are also in agreement with findings of [12] in which accent recognition performance improves with increment in the number of factors in the i-vector extraction system.

Table 1: *Performance of i-vector system in CallFriend corpus for selected i-vector dimensions (EER %). UBM has 1024 Gaussians.*

i-vector	English		Mandarin		Spanish	
	Gaussian scoring	Cosine scoring	Gaussian scoring	Cosine scoring	Gaussian scoring	Cosine scoring
200	22.14	23.20	20.12	20.49	20.43	20.87
400	21.81	22.60	18.70	19.11	19.10	20.21
600	20.04	21.30	17.84	18.45	18.38	19.63
800	18.54	19.83	15.74	16.31	16.56	18.63
1000	**17.18**	**18.01**	**14.77**	**14.91**	**15.00**	**16.01**

Table 2 shows the effect the UBM size. A curious observation is insensitivity of the i-vector performance to UBM size; the UBM with smaller size outperforms larger UBM. Also, Gaussian scoring outperforms cosine scoring as above.

The effect of varying the dimension of HLDA projection matrix is shown in Figure 2. The result suggests that reducing the dimensionality of i-vector considerably affects recognition accuracy. However, too aggressive reduction of i-vector dimensionality reduces accuracy. Although i-vectors of high dimension can be viewed as including more discriminatory variability, which then also contain more channel variability that degrades accuracy [18]. Dimensionality reduction result is comparable with findings in i-vector based speaker and language recognition systems, where applying LDA, a special case of HLDA, led to improvements in results [9, 27].

Table 2: *Performance of i-vector system in CallFriend corpus for five selected UBM sizes (EER %). i-vectors are of dimension 600.*

UBM	English		Mandarin		Spanish	
	Gaussian scoring	Cosine scoring	Gaussian scoring	Cosine scoring	Gaussian scoring	Cosine scoring
256	**20.01**	**21.12**	17.26	17.93	**18.30**	**19.00**
512	20.14	21.61	**17.20**	**17.91**	18.32	19.15
1024	20.04	21.30	17.84	18.45	18.38	19.63
2048	23.23	23.81	20.18	21.15	21.03	22.01
4096	23.90	23.89	20.23	21.57	21.84	22.66

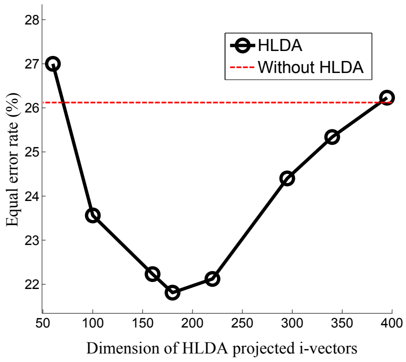

Figure 2: *Equal error rates at different dimensions of HLDA projected i-vectors in CallFriend corpus.*

As one of the aims in i-vector approach is to maximize the captured total variability, we also investigated the effect of changing the corpus in training the T-matrix. To this end, we used estimated sufficient statistics of the FSD corpus utterances for training the T-matrix in CallFriend corpus. Results are given in Table 3, where the only difference between rows is the corpus used to train the T-matrix. It should be noted that in case of CallFriend corpus, for a selected language, all training utterances of two other languages were used to train the T-matrix. As expected, the recognition accuracy increases when T-matrix is trained from the same corpus as the sufficient statistics have been computed.

Table 3: *Change of corpus in training the T-matrix in CallFriend corpus experiment (EER %). UBM is of size 1024 and i-vectors of dimension 600, Gaussian scoring.*

Corpus used for T-matrix	English	Mandarin	Spanish
CallFriend	**20.04**	**17.84**	**18.38**
FSD	23.77	22.30	22.81

Performance of i-vector system in foreign accent recognition experiment is shown in Table 4. It is interesting that foreign accent recognition seems to be more challenging than dialect recognition. The smallest EER achieved in FSD corpus is 16.56% compared to 14.77% best EER performance in Mandarin dialects of CallFriend corpus. For some accents such as Estonian, Kurdish and Russian, this difficulty is more pronounced. Linguistically, those languages close to Finnish are more difficult to be discriminated. Estonian is Uralic language as is Finnish. Kurdish and Russian, in turn, are an Indo-European languages, but do not belong to the same sub-family as English. Moreover, Speakers of a dialects in CallFriend are native speakers, so one can expect a uniform language speaking ability. But for speakers of a foreign language, accentedness is correlated with the ability to speak the target language (Finnish in this case). In conclusion, speech material from where the detections are made from.

Table 4: *Performance of i-vector system in FSD corpus (EER %). UBM is of size 256 and i-vectors of dimension 1000.*

Accents	# of Utterances	Gaussian scoring	Cosine scoring
Spanish	72	16.56	16.90
Turkish	100	16.63	16.37
Chinese	78	18.83	18.64
Albanian	85	18.86	18.89
English	106	19.44	21.03
Arabic	194	22.21	23.42
Russian	1061	25.28	26.76
Kurdish	93	25.50	27.31
Estonian	184	26.51	28.53
All	1973	20.01	22.00

Finally, in Table 5, regarding the optimal parameters achieved in the previous experiments, we demonstrate the best i-vector performance achieved so far and compare the results with the GMM-UBM system. The results indicate that i-vector system outperforms the conventional GMM-UBM system in both corpora as one expects, however, much more work is needed to include i-vector system. We believe that the i-vector performance reported in this work is not the best performance that i-vector system could achieve. As mentioned in [12, 13], relying on good back-end classifiers can considerably improve performance of i-vector system in accent recognition, but this is left as future work.

Table 5: *Comparison between best overall i-vector performance and GMM-UBM system in CallFriend and FSD corpora (EER %). UBM is of size 256, i-vectors of dimensionality 1000 and HLDA projected i-vectors of dimension 180, Gaussian scoring.*

Corpus	GMM-UBM	i-vector
CallFriend	18.73	**15.06**
FSD	24.13	**20.01**

5. Conclusions

In this paper, we have investigated the effectiveness of i-vector system in context of dialect and foreign accent recognition systems. Our findings demonstrate that i-vector system outperforms classic GMM-UBM as one expects. Foreign accent recognition is found more challenging than dialect recognition. We have also shown that i-vector performance is dependent to dimensionality of i-vectors, choices of corpus in training the T-matrix and dimension of projected i-vectors.

6. Acknowledgements

We would like to thank Ari Maijanen from University of Jyväskylä for an immense help with the FSD corpus. This work was partly supported by Academy of Finland (projects 253000 and 253120).

7. References

[1] J. Nerbonne, ”Linguistic variation and computation”. In Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics, pages 3–10, 2003.

[2] F. Biadsy, ”Automatic dialect and accent recognition and its application to speech recognition”, Columbia University, 2011.

[3] N.F. Chen, W. Shen and J.P. Campbell, ”A linguistically-informative approach to dialect recognition using dialect-discriminating context-dependent phonetic models” Acoustics Speech and Signal Processing (ICASSP), pages 5014-5017. 2010.

[4] P.A. Torres-Carrasquillo, T.P. Gleason and D.A. Reynolds, ”Dialect identification using Gaussian mixture models”, In Proceeding Odyssey: The Speaker and Language Recognition Workshop, pages 757–760, 2004.

[5] G. Liu and J.H. Hansen, ”A systematic strategy for robust automatic dialect identification”, In EUSIPCO2011, pages 2138–2141, 2011.

[6] M.A. Zissman, T.P. Gleason, D.M. Rekart and B.L. Losiewicz, ”Automatic Dialect identification of extemporaneous conversational latin american spanish speech”, International Conference on Acoustics, Speech, and Signal Processing, ICASSP, 1995.

[7] T. Wu, J. Duchateau, J. Martens and D. Compernolle, ”Feature subset selection for improved native accent identification”, Speech Communication, pages 83–98, 2010.

[8] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel and P. Ouellet, ”Front-end factor analysis for speaker verification”, IEEE Transactions on Audio, Speech and Language Processing, pages 788-798, 2011.

[9] A. Kanagasundaram, R. Vogt, D. Dean, S. Sridharan and M. Mason, ”i-vector based speaker recognition on short utterances”, In Interspeech, pages 2341–2344, 2011.

[10] D. Martinez, O. Plchot, L. Burget, O. Glembek and P. Matejka, ”Language recognition in ivectors space”, In Interspeech, pages 861–864, 2011.

[11] H. Li, B. Ma and K.A. Lee, ”Spoken language recognition: from fundamentals to practice”, In proceeding of Spoken Language Recognition, vol. 101, pages 1136–1159, 2013.

[12] A. DeMarco and S.J. Cox, ”Iterative classification Of regional british accents In i-vector space”, Symposium on Machine Learning in Speech and Language Processing (SIGML), 2012.

[13] M.H. Bahari, R. Saeidiy, H.V. hamme and D. van Leeuwen, ”Accent recognition using i-vector, Gaussain mean supervector, Guassian posterior probability for spontaneous telephone speech”, Accepted to in International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vancouver, Canada, 2013.

[14] M. Loog and R.P. Duin, ”Linear dimensionality reduction via a heteroscedastic extension of LDA: The Chernoff Criterion” IEEE Transactions On Pattern Analysis and Machine Intelligence (PAMI), Vol. 26, No. 6, 2004.

[15] N. Dehak, R. Dehak, P. Kenny, N. Brummer, P. Ouellet and P. Dumouchel, ”Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verication”, In Interspeech, pages 1559-1562. 2009.

[16] D. Matrouf, N. Scheffer, B. Fauve and J.F. Bonastre, ”A straightforward and efcient implementation of the factor analysis model for speaker verication”, In International Conference on Speech Communication and Technology, pages 1242–1245, 2007.

[17] M.J. Gales, ”Semi-tied covariance matrices for hidden markov models”, IEEE Transaction on Speech and Audio Processing, pages 272–281, 1999.

[18] W. Rao and M.W. Mak, ”Alleviating the small sample-size problem in i-vector based speaker verification”, Chinese Spoken Language Processing (ISCSLP), pages 335–339, 2012.

[19] E. Singer, P.A. Torres-Carrasquillo, D. Reynolds, A. McCree, F. Richardson, N. Dehak and D. Sturim, ”The mitll nist lre 2011 language recognition system”, In Odyssey: The Speaker and Language Recognition Workshop, pages 4994–4997, 2011.

[20] ”CallFriend corpus”, In Linguistic Data Consortium, 1996.

[21] ”Finnish national foreign language certificate corpus”, University of Jyvaskyla, Centre for Applied Language Studies. http://yki-korpus.jyu.fi

[22] L. Lee and R.C. Rose, ”Speaker normalization using efficient frequency warping procedures”, In Acoustics, Speech, and Signal Processing, pages 353–356, 1996.

[23] M.A. Kohler and M. Kennedy, ” Language identification using shifted delta cepstra”, In Circuits and Systems Symposium, pages 69–72, 2002.

[24] P.A. Torres-Carrasquillo, E. Singer, M.A. Kohler, R.J. Greene, D.A. Reynolds and J.R. Deller, Jr, ”Approaches to language identification using Gaussian mixture models and shifted delta cepstral features”, In Interspeech, pages 89–92, 2002.

[25] J. McLaughlin, D.A. Reynolds and T. Gleason, ”A study of computation speed-UPS of the GMM-UBM speaker recognition system”, In EuroSpeech, pages 1215–1218, 1999.

[26] P. Matejka, L. Burget, O. Glembek, P. Schwarz, V. Hubeika, M. Fapso, T. Mikolov, O. Plchot and J.H. Cernocky, ”BUT language recognition system for NIST 2007 evaluations”, Interspeech, pages 739–742, 2008.

[27] N. Dehak, P.A. Torres-Carrasquillo, D. Reynolds and R. Dehak, ”Language recognition via i-vectors and dimensionality reduction”, In Interspeech, pages 857–860, 2011.