# iVector Approach to Phonotactic Language Recognition

Mehdi Soufifar[1], Marcel Kockmann[2], Lukáš Burget[2], Oldřich Plchot[2], Ondřej Glembek[2] and Torbjørn Svendsen[1]

[1]Department of Electronics and Telecommunications, NTNU, Trondheim, Norway
[2]Brno University of Technology, Speech@FIT, Czech Republic

{soufifar,torbjorn}@iet.ntnu.no, {kockmann,iplchot,burget,glembek}@fit.vutbr.cz

## Abstract

This paper addresses a novel technique for representation and processing of n-gram counts in phonotactic language recognition (LRE): subspace multinomial modelling represents the vectors of n-gram counts by low dimensional vectors of coordinates in total variability subspace, called iVector. Two techniques for iVector scoring are tested: support vector machines (SVM), and logistic regression (LR). Using standard NIST LRE 2009 task as our evaluation set, the latter scoring approach was shown to outperform phonotactic LRE system based on direct SVM classification of n-gram count vectors. The proposed iVector paradigm also shows comparable results to previously proposed PCA-based phonotactic feature extraction.

**Index Terms**: language recognition, subspace modeling, multinomial distribution.

## 1. Introduction

Spoken language recognition (LRE) is the task of automatically determining the language of a spoken utterance. There are two main approaches for this task: acoustic and phonotactic. The focus of this paper is on the latter approach.

In a classical phonotactic LRE, the front-end is usually consist of a phone recognizer used to tokenize speech utterances into discrete events. Based on the produced phone sequence, n-gram counts can be extracted. The n-gram counts are then served as input to either a generative classifier (e.g. smoothed n-gram language model), or discriminative ones such as support vector machines (SVM) [1] or logistic regression (LR). In the latter case, we need to represent the n-gram counts by a fixed-length vector, whose length depends on the size of the phone inventory and grows exponentially with the order of the n-gram. This is considered as a bottleneck for the phonotactic solutions and generates a need for compact representation of the original n-gram counts. A good compact representation makes training of the classifiers and also the classification task faster and enables us to use more data, which can eventually improve the system performance.

In [2], discriminative selection of the n-grams was proposed to tackle this problem . Another approach relies on principal component analysis (PCA) to reduce dimensionality of the vector of n-gram counts to a lower-dimension [3, 4].

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. D10PC20015. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA or its Contracting Agent, the U.S. Department of the Interior. The work was also partly supported by Czech Ministry of Education project No. MSM0021630528, and Grant Agency of Czech Republic project No. GP102/09/P635.

Our paper introduces *iVectors* as a powerful tool for compact representation of n-gram statistics. Since their introduction in speaker recognition (SRE) [5], iVectors have been used very successfully. They were originally proposed for continuous features, based on Gaussian mixture models (GMM). Recently, the idea of iVector has been extended for discrete features: Kockmann et al. [6] proposed a subspace multinomial model to model a discrete representation of the prosodic features in SRE. Since we are dealing with another discrete representation of speech utterances (output of the phone recognizer), we explore the idea of using the same method to model n-gram counts. After being derived, the iVectors are used as feature vectors in the discriminative classifiers. We also study the performance of logistic regression (LR) and support vector machines (SVM) as the back-end classifier.

We analyze the performance of the proposed iVector paradigm with respect to the baseline, where the whole vector of the n-gram counts is used as an input to the discriminative classifiers [1]. We also compare performance of the iVector approach with PCA-based dimensionality reduction presented in [4]. The experiments are conducted on the NIST LRE 2009 task and all results are given in terms of the average decision cost function ($C_{avg}$) according to the NIST LRE2009 evaluation plan [7].

## 2. Subspace models

The basic assumption in subspace modeling is that the natural parameters of a model usually live in a much smaller subspace than the full parameter space. This subspace can be learned by introducing latent variables in the model.

### 2.1. iVectors based on continuous features

The classical formulation of Joint Factor Analysis (JFA) model for speaker verification [8] assumes that the super-vector of the concatenated mean vectors $\phi$ of a GMM are distributed according to a subspace model consisting of two separate subspaces for speaker (within-class) and channel variability:

$$\phi = \mathbf{m} + \mathbf{V}\mathbf{y} + \mathbf{U}\mathbf{x}, \tag{1}$$

where $\mathbf{m}$ is a speaker- and channel-independent supervector, and $\mathbf{V}$ and $\mathbf{U}$ span linear subspaces (for speaker and channel variability) in the original mean parameter space. The $\mathbf{y}$ and $\mathbf{x}$ are the low-dimensional latent variables corresponding to the speaker and channel subspaces.

A simplified variant of JFA [5] assumes that speaker and channel subspaces are not decoupled and uses only one sub-

space covering the total variability in an utterance:

$$\phi = \mathbf{m} + \mathbf{T}\mathbf{w}. \quad (2)$$

Again, $\mathbf{T}$ spans a linear subspace in the original mean super-vector space and $\mathbf{w}$ is the corresponding low-dimensional latent variable. The iVector is a point estimate of $\mathbf{w}$ obtained by adapting the model in (2) to a given utterance. Unlike JFA, the JFA-like model now serves only as the extractor of the vectors $\mathbf{w}$, which can be seen as low-dimensional fixed-size representations of utterances, which are in turn used as inputs to another classifier.

The extracted iVectors contain information about both the class of the utterance and the channel. In the context of SRE, we have many classes with limited amount of training data and the task is to make decision on whether two utterances belong to the same speaker or to different speakers. As a result, we need to model both speaker and channel variability in iVector space by means of a proper technique, usually probabilistic linear discriminant analysis (PLDA) [9]. However, in the context of LRE, we have a small number of classes and a relatively large amount of training data per class and we train discriminative classifier for each class.

### 2.2. iVectors based on multinomial distribution

Discrete features can also be modeled under the subspace paradigm. Discrete events can be modeled using multinomial distribution and similar to the continuous feature case, we can assume that there is a low-dimensional subspace of the parameter space in which the parameters of the multinomial distributions for individual utterances live. In the context of phonotactic LRE, every speech utterance can be represented by a fixed-length vector containing discrete n-gram statistics. The log-likelihood of $n^{th}$ utterance represented by $E$-dimensional vector of n-gram counts ($\nu_n$), can be calculated as

$$\log(P(\nu_n \mid \phi_n)) = \sum_{e=1}^{E} \nu_{ne} \log \phi_{ne}, \quad (3)$$

Where $\nu_{ne}$ is the occupation count for the n-gram $e$ and utterance $n$. The $\phi_{ne}$ is the utterance-dependent model parameter, representing probability for the corresponding n-gram. Log-likelihood of a set of utterances is given by

$$\sum_{n=1}^{N} \log P(\nu_n \mid \phi_n), \quad (4)$$

where $N$ is the number of utterances. The utterance-dependent model parameter $\phi_{ne}$ can be estimated by means of subspace modelling of the n-gram counts as follows:

$$\phi_{ne} = \frac{\exp(m_e + \mathbf{t}_e \mathbf{w}_n)}{\sum_{i=1}^{E} \exp(m_i + \mathbf{t}_i \mathbf{w}_n)}, \quad (5)$$

where $\mathbf{w_n}$ is an utterance-dependent latent variable and $\mathbf{t_e}$ is the $e^{th}$ row of the subspace matrix $\mathbf{T}$, which spans a linear subspace in the log-probability domain.

Given the parameters $\mathbf{m}$ and $\mathbf{T}$ we can estimate $\mathbf{w}$ to maximize the log-likelihood in (3) for the corresponding utterance. The estimated $\mathbf{w}$ is called iVector. Similar to the case of continuous features, the subspace multinomial model is used as a feature extractor and each iVector can be seen as a low-dimensional representation of the whole utterance.

### 2.3. Parameter estimation

The parameters of the model for iVector extraction are estimated using maximum likelihood (ML) estimation. To do so, the $\mathbf{T}$ matrix and $\mathbf{w}$(vectors for all training utterances) are estimated through an iterative algorithm, where we alternate between estimation of $\mathbf{w}$ with fixed bases ($\mathbf{T}$) and estimation of $\mathbf{T}$ with fixed $\mathbf{w}$. We keep the value of $\mathbf{m}$ fixed during estimation. Note that even by fixing one of $\mathbf{T}$ or $\mathbf{w}$, to estimate the other, there is no closed form solution for this problem. As a consequence, every iteration of parameter estimation needs to be done through a nonlinear optimization algorithm. In [6], a Newton Raphson algorithm with a variation of the Hessian matrix was proposed to estimate the model parameters of a similar model for the SRE task. The same formulation is used in this work. We initialize the $m_e$ as

$$m_e = \log(\frac{1}{N} \sum_{n=1}^{N} \nu_{ne}) \quad (6)$$

and the $\mathbf{T}$ matrix with small random numbers. Since this is a concave optimization problem, we can continue the iteration as long as the log-likelihood in (3) is increasing [10]. The trained $\mathbf{T}$ matrix is then used to extract iVectors for all train, development an test utterances.

Note that a similar model was already used for LRE in [11]. However, it was used to apply channel compensation similar to the JFA paradigm in the context of phonotactic LRE, while we are using it for extracting low-dimensional features. A similar model was also used for modelling of the weights in subspace GMM [12], which was served as inspiration to derive re-estimation formula in [6].

## 3. Experimental setup

In this section, we briefly go through experimental setup and each part of the baseline system and the PCA-based feature extraction. After that, we explain the proposed system.

### 3.1. Data

The original training data is divided into two sets denoted as TRAIN and DEV. The TRAIN set comprises data from 23 languages corresponding to the target language list of the NIST LRE09 task [7]. After that, we limit the TRAIN set to at most 500 utterances per language as proposed in [13], resulting in 9763 segments (345 hours of recording). This way, we have almost balanced amount of training data per language, which will avoid to bias a classifier toward a language with lots of training data. The DEV set contains 38469 segments from 23 languages according to the list of target languages in the NIST LRE09 task. The DEV set mainly consists of data from the previous NIST LRE tasks plus some extra longer segments from the standard conversational telephone speech (CTS) databases (CallFriend, Switchboard etc.) and voice of America (VOA) data. The TRAIN and the DEV sets contain disjoint set of speakers. This way, the system should learn the language of the utterance not the speaker. Full description of the used data is given in [14]. The evaluation data (EVAL set) is identical to what was provided by NIST for the NIST LRE 2009 task.

### 3.2. Vector of n-gram counts

The n-gram counts were extracted using the Hungarian phone recognizer (HU) from Brno University of Technology (BUT),

based on a hybrid ANN/HMM approach [15]. Only 3-gram counts were used in our system and neither 2-gram nor 1-gram improved the system performance. The 3-gram expected counts are extracted from phone lattices, generated by the HU phone recognizer. The Hungarian phoneme list contains 61 phonemes. We use the mapping proposed by MIT to merge down the Hungarian phone list to 33 phonemes. This results in $33^3 = 35937$ 3-grams. We also took square roots of the expected n-grams counts before going through other steps in all the systems. The square root compresses the dynamic range of the counts and slightly improves the performance over all systems.

### 3.3. Baseline setup

Our baseline is the BUT phonotactic LID system; as a part of the BUT-AGNITIO submission to the NIST LRE 2009 [14]. This system is denoted as BASE in the rest of this paper and uses the whole vector of n-gram counts as the input to the SVM classifiers. For this system, the expected counts are used to train 23 linear SVM classifiers in a one-to-all manner using LIBSVM tool[1] [1]. This system does not use any dimensionality reduction technique to reduce the size of 3-gram counts vectors. The scores are then passed to a calibration back-end, which is trained on the DEV set using jack-knifing scheme. The calibration back-end comprises linear generative model followed by a multi-class logistic regression (MLR) as described in [14].

### 3.4. PCA-based vector compression

We also implemented the PCA-based dimensionality reduction approach according to [4]. It is denote as PCA-BASE. In this system, a transformation matrix $\mathbf{U}$ is extracted from the TRAIN set. Using trained $\mathbf{U}$ matrix, all the utterances in TRAIN, DEV and EVAL sets are transformed to a lower dimension. The transformed TRAIN set is then used to train the discriminative classifiers (LR and SVM) in the same configuration as the BASE system.

### 3.5. Classifier

The low dimensionality of the iVectors and also the feature vectors generated by PCA makes it possible to use LR as the classifier. In the LRE task, we would like to have output scores of the classifiers in the form of log-likelihood ratio (LLR) and LR has this instinctive characteristic while SVM scores have no probabilistic interpretation [10]. The performances of LR and SVM as classifiers over DEV and EVAL sets are given in Table 1. The results are given for PCA-based vector compression with the target dimension of 600. As we were expecting, LR performs consistently better than SVM over all the conditions. Similar behavior was also observed for the proposed iVector feature extraction. As a result, LR is used as a classifier for the rest of the experiments. In fact, multi-class logistic regression (MLR) would fit this problem better. However, we used LR binary classifiers in the same configuration as the SVMs in the BASE system to keep the systems comparable.

### 3.6. Proposed system

In the proposed system, the 3-gram counts are produced as described in section (3.2). Applying the technique explained in the section (2.2), the vectors of the 3-gram counts are represented by the corresponding iVectors, yet, the number of iterations for estimating $\mathbf{T}$ and $\mathbf{w}$ should be decided. We could update the $\mathbf{T}$ matrix through the iterations as long as the model likelihood is increasing over the TRAIN set. However, we only continue the iterations as long as the likelihood on the DEV set is also increasing to avoid over-training. On the other hand, our final metric to assess the system performance is the $C_{avg}$. The final number of iterations was therefore decided based on the both criterions and it varies for various size of the n-gram counts and also targeted subspace dimensionality. Mostly, 5 iterations were chosen for the rest of the experiments.

Table 1: The $C_{avg} \times 100$ for PCA-BASE system with different classifiers on DEV and EVAL sets over all the conditions

| | DEV | | | EVAL | | |
|---|---|---|---|---|---|---|
| Classifier | 30s | 10s | 3s | 30s | 10s | 3s |
| SVM | 2.83 | 7.05 | 17.77 | 3.62 | 8.89 | **21.09** |
| **LR** | **2.22** | **6.22** | **17.26** | **2.93** | **8.29** | 22.60 |

The number of useful dimensions is believed to be much lower than the original dimensionality in the vector of n-gram counts [3, 4]. The performances of the proposed subspace model on DEV set for different sizes of the subspace are shown in Fiure 1. In fact, increasing dimensionality of the subspace would always result in longer training time and higher memory requirement and we are interested to have lower dimensional representation of the original feature vector. 600 dimensions seem to be an appropriate dimensionality since increasing the dimension to 800 or 1000 did not improve system performance on DEV set. Even though $C_{avg}$ on the DEV set is our criterion to decide on subspace dimensionality, $C_{avg}$ on the EVAL set for the corresponding dimensions are also included in Figure 1 to observe the system performance on an independent evaluation set.

Based on the findings in Table 1, we used LR as a classifier in the proposed system. The generated iVectors are used to train 23 linear logistic regression classifiers in one-to-all manner using LIBLINEAR package[2]. For calibration of the scores, the same back-end as in the BASE system was deployed.

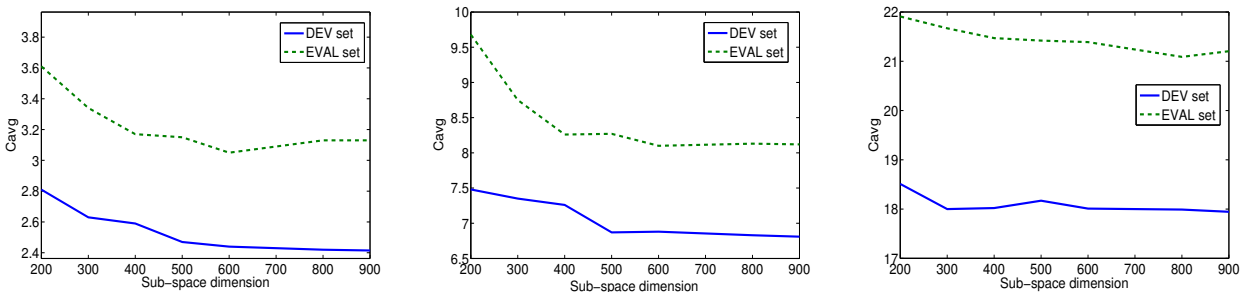
## 4. System evaluation & results

Based on Figure 1, we chose 600 dimensional subspaces trained with 5 iterations. All the systems are built based on the output of the phone recognizers. However, the phone recognizers do not detect any speech in some of the input utterances. In fact, the NIST LRE 2009 key file also shows that there is no speech part in some of the utterances. To deal with this, after calibration of the scores, we put 0 as the scores for those particular utterances, implying that our system could not decide on the language of the utterance.

The results for the base-line and the iVector feature extraction are given in Table 2. For PCA-BASE, the original 35937 dimensions is reduced to 600 using PCA as proposed in [4] and in BASE, no dimensionality reduction technique was used and the SVM classifiers were trained on the whole 35937 dimensions. Comparing the IVECT with the BASE system, not only we did not loose any system performance, but also slightly better results were achieved. Comparing the IVECT and PCA-BASE, the iVector feature extraction performs the same as PCA-based dimensionality reduction and is slightly better in the short con-

[1] http://www.csie.ntu.edu.tw/ cjlin/libsvm

[2] http://www.csie.ntu.edu.tw/ cjlin/liblinear




(a) 30 seconds (b) 10 seconds (c) 3 seconds

Figure 1: The $C_{avg} \times 100$ on DEV and EVAL set for different subspace dimensions over 30s, 10s and 3s conditions

Table 2: $C_{avg} \times 100$ for different systems on NIST LRE09 Evaluation task over 30s, 10s and 3s conditions.

| System | dimension | 30s | 10s | 3s |
|---|---|---|---|---|
| BASE | 35937 | 3.12 | 8.56 | 21.01 |
| PCA-BASE | 35937→600 | **2.93** | 8.29 | 22.60 |
| **IVECT** | 35937→600 | 3.05 | **8.10** | **21.39** |

ditions.

## 5. Discussions and Conclusions

We proposed a novel method to extract the iVectors by means of subspace multinomial modelling of the n-gram counts. Using the proposed subspace model, the huge vector of the n-gram counts are represented by the low-dimensional iVector while preserving the discriminative power of the vector. By using the iVectors, we earned slightly better results compared to the BASE, which does not use any dimensionality reduction. Even though these are just preliminary experiments on using the proposed method in LRE, it has shown comparable results to PCA-based feature extraction. Use of higher order n-gram statistics would most likely improve the results and it has to be explored.

We showed that, in the context of LRE task, logistic regression outperforms SVM as a classifier and it consistently improves the system performance. Since the iVector feature extraction shows acceptable performance with binary LR classifiers, we can expect to get even better performance by MLR as a more appropriate classifier.

In our approach, we assume that the n-gram counts represent discrete independent events drawn from a single multinomial distribution. A more appropriate approach would be to cluster the counts according to the n-gram histories and model each such cluster by a separate distribution. This approach, however, suffers from data sparsity and is yet to be dealt with as the future work for this approach.

## 6. References

[1] W. Campbell, F. Richardson, and D. Reynolds, "Language recognition with word lattices and support vector machines," in *Proc. of ICASSP*, Honolulu, Hawaii, USA, 2007.

[2] F. Richardson and W. Campbell, "Language recognition with discriminative keyword selection," in *Proc. of ICASSP*, Las Vegas, USA, 2008.

[3] H. Li, B. Ma, and C.-H. Lee, "A vector space modeling approach to spoken language identification," *ASLP, IEEE Transactions on*, vol. 15, no. 1, pp. 271 –284, 2007.

[4] T. Mikolov, O. Plchot, O. Glembek, P. Matejka, L. Burget, and J. Cernocky, "PCA-based feature extraction for phonotactic language recognition," in *Proc. of Odyssey*, Brno, CZ, 2010.

[5] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *ASLP, IEEE Transactions on*, vol. 19, no. 4, pp. 788 –798, 2011.

[6] M. Kockmann, L. Burget, O. Glembek, L. Ferrer, and J. Cernocky, "Prosodic speaker verification using subspace multinomial models with intersession compensation," in *Proc. of ICSPL*, Makuhari, Chiba, Japan, 2010.

[7] "NIST language recognition evaluation plan," 2009, http://www.itl.nist.gov/iad/mig/tests/lre/2009/.

[8] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification," *ASLP, IEEE Transactions on*, vol. 16, no. 5, pp. 980 –988, 2008.

[9] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," *Computer Vision, IEEE International Conference on*, vol. 0, pp. 1–8, 2007.

[10] C. Bishop, *Pattern recognition and machine learning.* Springer, 2006, ch. 4.

[11] O. Glembek, P. Matejka, L. Burget, and T. Mikolov, "Advances in phonotactic language recognition," in *Proc. of ICSLP*, Brisbane, AU, 2008.

[12] D. Povey, L. Burget, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. Goel, Karafiat, M., A. Rastrow, R. Rose, P. Schwarz, and S. Thomas, "Subspace Gaussian mixture models for speech recognition," in *ICASSP*, Dallas, US, 2010.

[13] N. Niko Brummer, L. Burget, O. Glembek, V. Hubeika, Z. Jancik, M. Karaat, P. Matejka, T. Mikolov, O. Plchot, and A. Strasheim, "But-agnitio system description for NIST LRE 2009," http://www.fit.vutbr.cz/research/view_pub.php.en?id=9551.

[14] Z. Jancik, O. Plchot, N. Brummer, L. Burget, O. Glembek, V. Hubeika, M. Karafiat, P. Matejka, T. Mikolov, A. Strasheim, and J. Cernocky, "Data selection and calibration issues in automatic language recognition - investigation with but-agnitio NIST lre 2009 system," in *Proc. of Odyssey*, Brno, CZ, 2010.

[15] P. Schwarz, P. Matejka, and J. Cernocky, "Hierarchical structures of neural networks for phoneme recognition," in *Proc. of ICASSP*, Toulouse, FR, 2006.