University of the Basque Country Systems for the NIST i-vector Machine Learning Challenge

Mikel Penagarikano, Mireia Diez, Luis J. Rodríguez-Fuentes,
Amparo Varona, Germán Bordel

GTTS Group,
Department of Electricity and Electronics
Faculty of Science and Technology
University of the Basque Country UPV/EHU,
48940, Leioa, Spain
e-mail: mikel.penagarikano@ehu.es

1 Introduction

This paper briefly describes the speaker recognition systems developed by the Software Technology Working Group (http://gtts.ehu.es) of the University of the Basque Country (UPV/EHU) for the NIST i-vector Machine Learning Challenge. Starting with the baseline system proposed by NIST, several variants were tried, some of them leading to performance improvements. The submitted system was finally based on the fusion of different subsystems.

2 The NIST i-vector Machine Learning Challenge

Unlike previous NIST Speaker Recognition evaluations, this challenge [1] tried to make the speaker recognition field accessible to participants from the machine learning community. This was somehow achieved by avoiding the participants to get access to the audio signal, but just to the i-vectors supplied by the organization and, thus, making the challenge feasible for researchers from outside the audio processing field.

Another difference with respect to previous SRE was the absence of a labelled development set. That is, along with the evaluation set, a large unlabelled development set was provided (i.e. the speaker identity of each i-vector was unknown), the set of development speakers being disjoint with the set of evaluation speakers. Maybe due to the unsupervised development scenario, the performance measure defined by NIST was based on the minimum value of a decision cost function, and therefore, the calibration loss was not penalized.

This work has been supported by the University of the Basque Country, under grant GIU13/28.

3 The EHU speaker recognition systems

3.1 NIST baseline system

NIST provided a baseline system with the following key features:

- Use the development set to center and whiten the evaluation i-vectors (i.e. a global mean i-vector and a global covariance matrix are estimated on the development set).
- Length normalize the i-vectors.
- Average each target speaker ivectors and project the averaged i-vectors again into the unit sphere.
- Compute trial scores in terms of dot-scoring (inner product).

3.2 Study of variability directions

A first effort was focused on analyzing the effect of removing some of the variability directions of the data. Instead of using all the 600 eigenvectors of the covariance matrix to build a square whitening matrix, only some of them were used (i.e. the i-vector dimension was reduced). Table 1 shows the results when different ranges of eigenvectors were used. Experiments showed that the speaker identity information was mainny located in the first half of the main variability directions. In fact, discarding the lowest 300 variability directions slightly outperformed the baseline score.

System ID	I-vector size	Score (progress set)
Baseline (W1:600)	600	**0.386**
W1:500	500	0.384
W1:400	400	0.379
W1:300	300	**0.376**
W1:200	200	0.390
W101:500	400	0.499
W201:400	200	0.710
W201:600	400	0.630
W401:600	200	0.847

Table 1: Results scored on the progress set for different whitening methods, where the system id *WX:Y* states for doing the whitening using only the eigenvectors in the range X:Y (eigenvectors were sorted in descending eigenvalue order). The Baseline system corresponds to using all the eigenvectors (i.e. W1:600).

3.3 Logistic Regression based scoring

Next, the scoring method was improved. Instead of using a simple inner product, a discriminative Logistic Regression model was trained for each target speaker using a one-versus-all configuration, where the five i-vectors of the target speaker were faced to all the i-vectors in the development set. The Bilbao toolkit [2] was used to estimate logistic regression models based on flat priors. Adding regularization to the model estimations did not improve the unregularized result, as shown in Table 2.

Although the scoring method was far more complex than the baseline one, the full experiment took just about one hour long.

System ID	Regularization parameter	Score (progress set)
W1:300 + LR	0	**0.326**
W1:300 + LR	0.01	0.330
W1:300 + LR	0.1	0.334
W1:300 + LR	1	0.343

Table 2: Results scored on the progress set for Logistic Regression modeling and using different regularization parameters.

3.4 Neural Networks

The Bilbao Toolkit contains some functionalities to estimate single hidden layer neural network based multiclass classifiers. Equivalently to what was done in the logistic regression modeling, a neural network was trained for each target speaker using a one-versus-all configuration (using development set speakers as impostors). Unlike with the logistic regression models, no improvement was obtained when using neural networks, despite being tried many different configurations. For sure, a more deep insight into the implemented neural networks would have led us to more satisfactory results.

3.5 Unsupervised within-class covariance estimation

The absence of a labelled development set proved to be a serious challenge in order to apply other classification technologies, since many of them rely on the estimation of a common within-class covariance matrix. For testing the goodness of the covariance estimation, the full covariance whitening transformation of the baseline system was replaced by an LDA transformation, which should theoretically outperform the baseline result. Two different strategies were tested:

Indirect estimation based on an unsupervised clustering.

The unsupervised clustering faces two main problems. First, a distance metric is needed. Using the euclidean distance in the baseline space implies feeding back the classification errors we are trying to avoid. Second, the number of classes in the development set must be guessed. Taking it to the limit, estimating only two classes in the development set could led us to indirectly estimate the within-gender covariance matrix.

Different methods were used to perform the unsupervised clustering: K-means clustering using euclidean and inner product distances and a GMM based soft estimation of the common within-class covariance matrix. Different number of classes (in the range from 500 to 5000) were used, but none of them outperformed the baseline approach.

Direct estimation from data.

Given a set of i-vectors pairs coming from the same speaker (note that pairs come from different speakers, but the two i-vectors in a pair come from the same speaker), then the difference of i-vector pairs follows [3]:

$$[x - y] \sim \mathcal{N}(0, 2C)$$

where C is the common within-class covariance matrix. For each i-vector of the development set, the N closest i-vectors would be used to estimate the covariance matrix. Note that, for example, the logistic regression model of Subsection 3.3 could be used to search for the closest ivectors in the whitened 300-dimensional space, but the covariance matrix could be estimated in the original 600-dimensional space. Following this method, a litle improvent (not comparable to the logistic regression) was obtained.

3.6 Complementarity of Logistic Regression configurations

As the logistic regression model probed to be the most relevant method, some different configurations of this method were tried, searching for complementarity information that could be exploited by a simple fusion. Although discarding the 300 lowest variability eigenvectors probed to be the best configuration, a simple fusion of two systems (W1:600+LR and W1:300+LR) proved to improve the result. Therefore, a fusion of six different systems was tried: W1:600+LR, W1:500+LR, W1:400+LR, W1:300+LR, W1:200+LR and W1:100+LR. In the absence of a labeled development set, trying to turning in a fusion did not seem very promising, so a fused system based on the sum of the scores was first submitted, and then, the fusion tuning was manually tried against the progress set. Surprisingly, none of the changes applied to the initial flat weights did improve the result. This could be explained by the fact that the logistic regression calibrated the scores.

Table 3 sumarizes the main systems presented in this paper and the final evaluation result of the submitted system.

System ID	Score (progress set)	Score (evaluation set)
Baseline	0.386	—
W1:300	0.376	—
W1:300 + LR	0.326	—
Fusion	**0.302**	**0.294**

Table 3: Results scored on the progress set for the baseline, 300 dimensional whitening, the logistic regression modeling and the fusion of 6 logistic regression systems based on different dimensional whitening. The final fused system is also scored on the evaluation set.

References

[1] "The 2013-2014 speaker recognition i-vector machine learning challenge," Tech. Rep., 2013, `http://www.nist.gov/itl/iad/mig/upload/sre-ivectorchallenge_2013-11-18_r0.pdf`.

[2] Niko Brümmer, "Bilbao toolkit: Matlab tools for multiclass classification," Tech. Rep., 2012, `https://sites.google.com/site/bilbaotoolkit/`.

[3] Niko Brümmer, "A farewell to svm: Bayes factor speaker detection in supervector space," Tech. Rep., 2006, `http://sites.google.com/site/nikobrummer`.