



# Channel Detectors for System Fusion in the Context of NIST LRE 2009

Florian Verdet<sup>1,2</sup>, Driss Matrouf<sup>1</sup>  
 Jean-François Bonastre<sup>1</sup>, Jean Hennebert<sup>2</sup>

<sup>1</sup>Université d'Avignon et des Pays de Vaucluse, Laboratoire Informatique d'Avignon, France

<sup>2</sup>Département d'Informatique, Université de Fribourg, Fribourg, Switzerland

florian.verdet@univ-avignon.fr, driss.matrouf@univ-avignon.fr,  
 jean-francois.bonastre@univ-avignon.fr, jean.hennebert@unifr.ch

## Abstract

One of the difficulties in Language Recognition is the variability of the speech signal due to speakers and channels. If channel mismatch is too big and when different categories of channels can be identified, one possibility is to build a specific language recognition system for each category and then to fuse them together. This article uses a system selector that takes, for each utterance, the scores of one of the channel-category dependent systems. This selection is guided by a channel detector. We analyze different ways to design such channel detectors: based on cepstral features or on the Factor Analysis channel variability term. The systems are evaluated in the context of NIST's LRE 2009 and run at 1.65%  $minC_{avg}$  for a subset of 8 languages and at 3.85%  $minC_{avg}$  for the 23 language protocol.

**Index Terms:** language recognition, channel, channel category, fusion, factor analysis, channel detector.

## 1. Introduction

Automatic language recognition is the process of recognizing the language used in a sample of speech. Such systems can be evaluated in identification mode, electing a given language from a set of  $N$  languages, or, as for the work presented here, in verification mode, detecting if a candidate language is used in the input waveform. Significant progress has been made over the last decades at different levels of information such as the acoustic level [1, 2, 3, 4] and the phonotactic level [5, 1, 2]. Significant development has been stimulated by systematic comparisons of systems through evaluation campaigns such as the NIST Language Recognition Evaluations (LRE) in 1996, 2003, 2005, 2007 and 2009 [6].

A recurrent difficulty is the fact that the speech signal includes all sort of information that is not relevant to the task of language recognition – such as speaker and channel dependent information. Speaker variability includes biometrics, emotion and health while channel variability is contributed by different means of audio acquisition and transmission, such as background noise, microphone, transmission channel and encoding. Furthermore, this non-useful information usually varies from session to session. Here we refer to this as *session dependent*.

The feature extraction and modeling strategy should attempt to focus on the language dependent information while minimizing the effect of the speaker and session dependent information. There has been considerable progress on different normalization techniques to achieve this in the feature extraction step (e.g. [7]) and the modeling of acoustic features with session compensation [8]. After solid improvements in speaker verification [9, 10], the Factor Analysis (FA) approach to ses-

sion compensation also shows its usefulness in language recognition [4]. It uses a large set of training data consisting of many speakers and having session information that covers the one used in the testing conditions. Despite all these efforts, we still observe considerable sensitivity of the models to a mismatch of channels [11]. For instance, NIST's Language Recognition Evaluation 2009 [6] is carried out on data of two rather different channel categories, namely the traditional Conversational Telephone Speech (CTS) and phone bandwidth segments of radio broadcasts (Voice Of America, VOA).

If we have rather different channel categories, one possibility is to handle them separately and merge the channel-category dependent systems at a later stage. This leads to the idea of modelling the CTS and VOA conditions separately. The fusion may then be done at different levels [11]: 1) Pooling all data together since the beginning (having thus just one common system), 2) stacking channel-category dependent FA session compensation matrices in order to have a matrix with a CTS specific and a VOA specific part or 3) merging two completely channel-dependent systems only at the score level.

A simple, but nevertheless effective way to merge such systems at score level is using a system selector. This means that, for each test utterance, the scores of one or the other system are taken (selected). Typically, such a system selector acts according to the channel category detected in the test utterance and selects the scores of the corresponding system. The work presented here investigates different ways to design channel detectors in the context of such a system selector. More specifically, we design channel detectors based on (shifted delta) cepstral features, as well as on Factor Analysis level, where the term containing the session and channel variability is used. This is a simple, but novel idea, which at the same time also validates the fundamental idea behind the FA approach.

Section 2 gives an overview of our FA systems and a description of how they are evaluated. In Section 3, we sketch the training and testing data. The different channel detectors under analysis are introduced in Section 4. The results obtained are presented in Section 5, followed by conclusions and some perspectives in Section 6.

## 2. GMM-UBM system with Factor Analysis

The first step of our training procedure is to compute a so-called *Universal Background Model* (UBM), which in our case is a language independent Gaussian Mixture Model (GMM). The parameters of the model are estimated using a standard Expectation Maximization (EM) algorithm by taking as much and as diverse data as possible from a large set of languages.

## 2.1. GMM-UBM with Factor Analysis

Factor Analysis works in a super-vector space where  $m_{ubm}$  is the super-vector (SV) composed of the mean vectors of the Gaussian mixtures concatenated together [10]. The basic Factor Analysis (FA) formula can be stated as:

$$m_{observed} = m_{ubm} + Dy_{language} + Ux_{session} \quad (1)$$

where  $m_{observed}$  is the super-vector of expected means of the observed data according to the UBM,  $Dy$  is the language specific term, and  $Ux$  represents the session variability, which is included in the observed data and which has to be discarded for the language model. The language dependent contribution  $y$  is weighted by a language independent diagonal matrix  $D$ . Factor Analysis assumes that the session dependent vector  $x$  is located in a lower-dimensional subspace which is projected back to super-vector space by the session compensation matrix  $U$  which is rectangular and session and language independent.

Each utterance is thus decomposed into the global part ( $m_{ubm}$ ), a language specificity ( $Dy$ ) and some session variability ( $Ux$ ). Another way to express this is that  $m_{ubm}$  is the centroids of all training data,  $Dy$  is an averaged offset from these centroids for each language and  $Ux$  is the residue corresponding to the session variability inherent to every single utterance.

### 2.1.1. Training of the FA parameters

The session compensation matrix  $U$  is common to all languages. It is iteratively estimated using the EM algorithm. Each step, the different  $x_{session}$  (variability) vectors are estimated, then a  $y_{language}$  is estimated for each language (using the new  $x$ ) and finally  $U$  is estimated globally, based on these  $x$  and  $y$ . Since  $x$  and  $y$  also depend on  $U$ , the process is iterated until convergence. The step by step algorithm is described in [10].

At the last iteration, the  $m + Dy_{language}$  part of the factor analysis formula (1) is injected back into the UBM as the Gaussian means to form the language model. Mixture weights and covariances are taken unchanged from the UBM. In other words, this last step corresponds somehow to a language specific MAP adaptation using session-compensated data.

### 2.1.2. Testing using compensated models

We also apply session compensation at the testing stage. There are two strategies for doing so, namely by removing the session contribution from the acoustic vectors or by moving the model parameters towards the unclean data hence injecting back the  $Ux$  term to the previously stored language model. We underline here the fact that  $x$  is estimated using statistics of the testing utterance obtained through the UBM. We therefore have language models where parameters are changing from test utterance to test utterance.

### 2.1.3. Channel-category dependent compensation

In the case different channel categories can be distinguished, the Factor Analysis approach described above can be applied in a category dependent way. These channel-category dependent systems can then be fused – for instance using a system selector which takes the output scores of the system that corresponds to the detected category [11].

For this article, two different types of such channel dependent systems are used in order to evaluate the effect of different channel detectors. For the *pure channel-category systems*, the compensation matrix  $U$  has a rank of 40 and is estimated exclusively on data of one channel category. The *merged-U systems*

use a common  $U$  matrix, which is obtained by stacking the two channel dependent  $U$  matrices and has thus a rank of 80.

## 2.2. Scoring and evaluation

Scores are normalized separately for each test utterance. Each likelihood score is divided by the maximum of the scores the utterance obtained against all language models.<sup>1</sup>

System performance is measured using *minimal average cost* ( $minC_{avg}$ ). It is the detection system choosing the decision threshold in such a way that the average expected cost of misses (utterances not recognized as being of the true language) and false acceptances (mistakenly detecting the presence of a language) among all target/non-target language pairs is minimal (see Section 4.1f of the LRE 2009 plan [6] for a description).

In our case, a false negative (a miss) and a false positive (false acceptance) decision have the same cost and the prior of a target trial is 0.5. The cost function that will be minimized is thus:

$$C_{avg} = \frac{1}{|L_T|} \sum_{l \in L_T} \left[ 0.5 \cdot P_{Miss}(l) + \frac{0.5}{|L_M|-1} \sum_{k \neq l \in L_M} P_{FA}(l, k) \right] \quad (2)$$

where  $L_T$  is the set of languages in the test data set (also called *target languages*),  $L_M$  is the set of languages for which we have models (*non-target languages*),  $P_{Miss}$  is the probability that a language model misses a match and  $P_{FA}(l, k)$  is the probability that an utterance of language  $l$  is mistakenly recognized as being of language  $k$ . It is thus the mean over all target languages of its probability to be missed and its average probability to be detected by a false language model.

## 3. Data parts

NIST LRE 2009 comprises 23 languages [6]. This article will not only evaluate the systems on the entire 30 second part, but also separately for the CTS and the VOA conditions. On the training side, we have CTS data available for 11 languages only<sup>2</sup> and VOA data for 22 languages<sup>3</sup>.

The fact that we do not have training data of both conditions for every language poses some troubles for training channel-category specific systems. When corresponding data is available, the language models are trained using this data. For the languages which are missing category-specific data, the training data of the other category is used. A more detailed description of the data sources may be found in [11].

### 3.1. Training data

Training material for the CTS condition is drawn from various sources: 1) All three parts of the CallFriend corpus for 8 languages (three of them in two dialects) with about 20 hours of speech per language/dialect, 2) the Indian English recordings with a nominal duration of 10 and 30 seconds of LRE 2005 development data, 3) the full conversations of the LRE 2007 evaluation data for 9 languages, and 3) the 10 and 30 second

<sup>1</sup>The reader knowing our prior works [4, 11] may have noticed that this is a change in normalization strategy. We looked for an even more simpler scheme than the division by the sum of the scores against all models, put to a power of  $K$ . This simple division does not depend on any tunable parameters, nor on the availability of a separate calibration data set (as required for more evolved backends), but still performs well.

<sup>2</sup>Spanish, English, Korean, Mandarin, Hindi, Indian English, Cantonese, French, Persian (Farsi), Russian and Vietnamese.

<sup>3</sup>The language missing VOA data being Indian English.

evaluation segments of LRE 2005 for 6 languages. Each language has between 40 and 2253 segments representing between 2.7 and 64.8 hours of speech. In total for 11 languages, we have 337 hours in 7870 segments.

The data of the VOA condition is drawn from the Voice Of America 3 (VOA3) data set<sup>4</sup> by limiting the number of utterances to a maximum of 400 for each language. For every language, they sum up to 3.0 to 27.9 hours of speech, hence a total of 333 hours across 8632 segments and 22 languages.

### 3.2. Testing data

Tests are conducted on NIST LRE 2009 data [6]. This evaluation set is composed of 41 794 utterances each containing nominally 3, 10 and 30 seconds of speech. The primary condition aggregates nur utterances of the 23 languages (closed-set). We focus only on the 30 second ones which comes down to 10 571 files giving that many target trials and thus 232 562 non-target trials. There are between 315 and 1015 testing files per language. From these testing files, 8708 are of CTS condition (10 languages) and 7490 of the VOA condition (22 languages).

## 4. Channel detector descriptions

The system selector for merging channel-category dependent systems bases its decisions (the scores of which system to take) on the advice of a channel detector. This section shows different designs for such a channel detector.

### 4.1. Simple sum

The *simple sum* fusion is not a channel detector, but a baseline replacement for the system selector. For each test, the scores of both channel-category dependent systems are summed together (without special weighting). This gives a minimal system performance which acts as a baseline for the different channel detectors.

### 4.2. Feature-based MAP

As a first approach, a Maximum A-Posteriori (MAP [12]) adapted model is estimated for each of the two channels using all training data (feature vectors) of that channel. Since the same UBM as for training the channel-dependent systems is used, these channel models are mixtures of 2048 Gaussians. This *f-MAP* detector has a channel identification rate of 87.63% when evaluated on the 30 second NIST LRE 2009 segments.

### 4.3. SVM on channel variability

Since Factor Analysis tries to model the session and channel variability separately and expressively, it is obvious to try to use this information for a channel detector. The channel variability part of the factor analysis formula (1) is the term  $Ux$ . Since  $U$  is fixed, the vector  $x$  represents the channel variability.

These  $x$  vectors (here with a dimension of 40) can directly be used as input SVs for a SVM [2, 3, 4]. The  $x$  vectors of the target category are taken as positive SVs and the  $x$  vectors of the other category as blacklist (negative examples). We notice that the SVMs we get for our two-category case are symmetric (in theory just the sign changes). This *x-SVM* detector has a channel identification rate of 87.41% on LRE 2009 30 seconds.

<sup>4</sup>LDC2009E40 (which also includes the VOA2 set).

### 4.4. MAP on channel variability

These FA  $x$  vectors can also be used as new features (front-end) on which a new channel-UBM can be estimated. This can then be adapted through MAP to obtain channel-dependent models working on these  $x$  vectors. For the work presented here, we use models of 64 mixtures (since each utterance is represented by one frame only). This *x-MAP* channel detector returns the channel of to the model with the bigger likelihood and has an accuracy of 75.29% on the 30 second LRE 2009 segments.

### 4.5. Oracle

The *oracle* represents the error-less channel detector. It returns the true channel category of an utterance. Evaluating the systems using the oracle as channel detector, gives the performance we want to approach by automatic channel detectors.

The performances of data based channel detectors are thus expected to lay between the one of a simple-sum fusion and that of the oracle.

## 5. Results

The parametric features used in this work are *Shifted Delta Cepstra* (SDC) in the configuration 7-1-3-7 [1, 3, 7] with energy based speech detection and mean/variance normalization, as described more in detail in [11]. All reported experiments are conducted using the free software framework MISTRAL<sup>5</sup>.

### 5.1. Evaluation on 8 common languages

Since there are some language-channel combinations which lack training or testing data, this section evaluates the systems on the NIST LRE 2009 30-second segments of the 8 common languages<sup>6</sup> only, as well as solely on the CTS and solely on the VOA subset.

#### 5.1.1. Pure systems

Table 1 presents the results of the two pure channel-category dependent systems and their fusion. The results of all automatic channel detectors fall in between those of a simple-sum fusion and the oracle. We observe that the best results among the automatic channel detectors are obtained by the *x-SVM* detector. They approach the ones of the oracle, which represents ground truth, quite well ( $\sim 6\%$  relative). The weakest channel detector is the one where the same  $x$  vectors are modeled by MAP.

Table 1: 8 languages, pure per-channel systems, in %  $\min C_{avg}$

base system	fusion	LRE 2009 closed-set 30s tests		
		all 30s	CTS only	VOA only
CTS	—	2.34	2.11	2.94
VOA	—	6.34	9.88	1.28
—	sSum	2.58	3.67	1.30
—	oracle	1.63	2.11	1.28
—	f-MAP	1.88	2.49	1.38
—	x-MAP	2.31	3.06	1.73
—	x-SVM	1.73	2.27	1.26

#### 5.1.2. Systems with merged-U matrix

The results shown in Table 2 are obtained by systems featuring a common stacked  $U$  matrix. The observations for the merged-U

<sup>5</sup>The MISTRAL project, <http://mistral.univ-avignon.fr>

<sup>6</sup>Cantonese, English, Hindi, Korean, Mandarin, Persian, Russian and Vietnamese; see [11] for more details.

systems are similar to those for the pure systems, with slightly better performances (4.3% relative gain for the  $x$ -SVM channel detector), except for the VOA only evaluation. When evaluated on channel-categories separately, the feature based MAP channel detector ( $f$ -MAP) is slightly better than  $x$ -SVM with 2.30%  $minC_{avg}$  for CTS tests and 1.44%  $minC_{avg}$  for VOA.

Table 2: 8 languages, merged- $U$  systems, in %  $minC_{avg}$

base system	fusion	LRE 2009 closed-set 30s tests		
		all 30s	CTS only	VOA only
CTS	—	2.53	2.04	3.63
VOA	—	6.30	6.64	1.48
—	sSum	2.40	3.48	1.60
—	oracle	1.55	2.04	1.48
—	$f$ -MAP	1.75	2.30	1.44
—	$x$ -MAP	2.27	2.96	1.87
—	$x$ -SVM	1.65	2.39	1.45

## 5.2. Evaluation on all 23 languages

This section presents the same systems under the 23 language NIST LRE 2009 condition. It also shows to which extent the systems are robust enough to recognize languages of a channel category for which no training data is available.

### 5.2.1. Pure systems

The results in Table 3 show that the automatic channel detectors achieve results that are a bit further off the oracle (about 12% relative) compared to the 8-language protocol, but they remain closer to the oracle than to the simple-sum performance.

Table 3: 23 languages, channel-dependent  $U$ , in %  $minC_{avg}$

base system	fusion	LRE 2009 closed-set 30s tests		
		all 30s	CTS only	VOA only
CTS	—	9.87	7.44	11.05
VOA	—	8.59	25.40	3.73
—	sSum	8.70	16.73	6.39
—	oracle	3.95	7.44	3.73
—	$f$ -MAP	4.47	8.24	4.02
—	$x$ -MAP	5.94	9.84	5.51
—	$x$ -SVM	4.65	8.35	4.36

### 5.2.2. Systems with merged- $U$

The performances of the systems using a common stacked  $U$  matrix are given in Table 4. They also indicate that these systems perform better than the pure systems. For the  $f$ -MAP channel detector, which, with 3.85%  $minC_{avg}$ , performs best, the enhancement over the channel-category dependent  $U$  matrix (pure) systems is 14% relative.

Table 4: 23 languages, merged- $U$  systems, in %  $minC_{avg}$

base system	fusion	LRE 2009 closed-set 30s tests		
		all 30s	CTS only	VOA only
CTS	—	6.59	6.63	7.51
VOA	—	5.80	13.77	3.43
—	sSum	4.32	7.17	3.92
—	oracle	3.64	6.63	3.43
—	$f$ -MAP	3.85	7.01	3.54
—	$x$ -MAP	4.78	7.58	4.51
—	$x$ -SVM	4.44	7.58	4.09

## 6. Conclusions and perspectives

The results show that channel detectors may be designed in different ways and that they may approach the performance of oracle based ground truth fusion up to 5-6% relative. Of the analyzed channel detectors,  $x$ -SVM and  $f$ -MAP achieve similar performances. Whereas the former performs marginally better when training data for all channel-categories (and languages) is available and the latter seems more robust to lack of such data. The results on the  $x$  vector based detectors confirm the basic idea behind Factor Analysis, in which the channel variability is captured by the  $Ux$  term of Formula (1).

The validation of  $x$  vector based channel detectors opens the interesting perspective of fully data based systems that automatically cluster and identify channel categories in the training data (instead of having labeled CTS and VOA). This is not possible on the feature level, since the information about the channel is mixed up with the information about the language.

## 7. References

- [1] Torres-Carrasquillo, P. A., Singer, E., Kohler, M. A., Greene, R. J., Reynolds, D. A. and Deller Jr., J. R., "Approaches to Language Identification Using Gaussian Mixture Models and Shifted Delta Cepstral Features", in Proc. International Conference on Spoken Language Processing in Denver, Colorado, ISCA, pp.82–92, September 2002.
- [2] Singer, E., Torres-Carrasquillo, P.A., Gleason, T.P., Campbell, W.M. and Reynolds, D.A., "Acoustic, phonetic, and discriminative approaches to automatic language identification", in Proc. of Eurospeech, pp.1345–1348, September 2003.
- [3] Campbell, W. M., Singer, E., Torres-Carrasquillo, P. A. and Reynolds, D. A., "Language Recognition with Support Vector Machines", in Proc. Odyssey: The Speaker and Language Recognition Workshop in Toledo, Spain, ISCA, pp.41–44, June 2004.
- [4] Verdet, F., Matrouf, D., Bonastre, J.-F. and Hennebert J., "Factor Analysis and SVM for Language Recognition", in Proc. of Interspeech '09, pp.164–167, Brighton, UK, 2009.
- [5] Zissman, M. A., "Comparison of four approaches to automatic language identification of telephone speech", Speech and Audio Processing, IEEE Transactions on, vol.4, no.1, pp.31–44, 1996.
- [6] The 2009 NIST Language Recognition Evaluation, evaluation plan, <http://www.itl.nist.gov/iad/mig/tests/lre/2009>
- [7] Matějka, P. and Burget, L. and Schwarz, P. and Černocký, J., "Brno University of Technology System for NIST 2005 Language Recognition Evaluation", in Proc. of Odyssey: The Speaker and Language Recognition Workshop, San Juan, PR, pp.57–64, 2006.
- [8] Castaldo, F., Colibro, D., Dalmaso, E., Laface, P. and Vair, C., "Compensation of nuisance factors for speaker and language recognition", in Audio, Speech, and Language Processing, IEEE Transactions on, vol.15, no.7, pp.1969–1978, September 2007.
- [9] Kenny, P., Boulianne, G., Ouellet, P. and Dumouchel, P., "Factor Analysis Simplified", in Proc. of ICASSP '05., vol.1, pp.637–640, March 18–23, 2005.
- [10] Matrouf, D., Scheffer, N., Fauve, B. and Bonastre, J.-F., "A straightforward and efficient implementation of the factor analysis model for speaker verification", in Proc. of Interspeech 2007, pp.1242–1245, 2007.
- [11] Verdet, F., Matrouf, D., Bonastre, J.-F. and Hennebert J., "Coping with Two Different Transmission Channels in Language Recognition", in Proc. of Odyssey 2010: The Speaker and Language Recognition Workshop, Brno, CZ, June 28–July 1, 2010. (currently in print – in the meantime it is accessible at <http://florian.verdet.ch/tmp/odyssey2010verdet.pdf>).
- [12] Gauvain, J.-L. and Lee, C.-H., "Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains", Speech and Audio Processing, IEEE Transactions on, vol.2, no.2, pp.291–298, April 1994.