The L^{2}F Language Recognition System for Albayzin 2012 Evaluation

Alberto Abad

L^{2}F - Spoken Language Systems Lab, INESC-ID Lisboa,
alberto@l2f.inesc-id.pt,
WWW home page: http://www.l2f.inesc-id.pt

Abstract. This document presents a description of INESC-ID's Spoken Language Systems Laboratory (L^{2}F) systems submitted to the Albayzin 2012 Language Recognition evaluation. The submitted systems differ on the number of sub-systems selected for fusion and the back-end configuration. The basic set of sub-systems considered are four conventional phonotactic sub-systems based on n-gram modelling of phoneme sequences, four additional phonotactic sub-systems based on SVM discriminative modelling of expected phone counts extracted from lattices, and an i-vector based sub-system with linear generative classifiers. Similarly to the L^{2}F Albayzin 2010 system, individual language models for clean and noisy conditions have been trained for each target language of the *Plenty of Training* condition. The L^{2}F *primary* system exploits Gaussian back-ends for each sub-system and linear logistic regression fusion of k sub-systems, selected automatically following a non-exhaustive fast greedy search method to find the best (sub-optimal) combination. This search process and the determination of the back-end parameters is performed per evaluation condition. Additionally, three contrastive systems have been developed. Language detection results have been submitted for all the evaluation conditions for every system.

Keywords: language recognition, Albayzin evaluations

1 Introduction

The "Red Temática en Tecnologías del Habla" (RTTH) has organised in the recent years a series of evaluations - so called Albayzin evaluations - in some relevant speech processing topics devoted to encourage language research activities on the four official languages of Spain. Similar to the well-known NIST Language Recognition Evaluation, a series of Language Recognition (LR) tasks have been proposed in 2008 and 2010. In the new Albayzin 2012 Language Recognition Evaluation there are significant novelties with respect to the previous editions. In contrast to previous campaigns, this year test data is considerably more challenging and consists of audios extracted from Youtube videos. Moreover, two different evaluation conditions have been proposed: *Plenty of Training* and *Empty Training*. For the *Plenty* condition, training data is provided for the target languages (Castilian, Catalan, Basque, Galician, Portuguese and English) and it can be used to train language models like in previous evaluation editions. In the new *Empty* condition, training data for the target languages (French, German, Greek and Italian) is not provided. In both cases, it is not allowed to use additional data from external sources for the development of the LR systems. Moreover, like in previous campaigns, *closed-set* and *open-set* conditions are also defined, resulting in a total of four possible evaluation conditions: *Plenty-Closed* (PC), *Plenty-Open* (PO), *Empty-Closed* (EC) and *Empty-Open* (EO). Detailed information on the evaluation campaign can be found in the evaluation plan document [1].

This document presents the LR systems developed by INESC-ID's Spoken Language Systems Laboratory (L^{2}F) for the Albayzin 2012 campaign. LR approaches can generally be classified according to the kind of source of information that they rely on. The most successful systems are based on the exploitation of the acoustic phonetics, that is the acoustic characteristics of each language, or the phonotactics which are the rules that govern the phone combinations in a language. Usually, the combination of different sources of knowledge and systems of different characteristics tends to provide increased language recognition performances [2]. For this evaluation, nine sub-systems have been developed: four phonotactic based on Phone Recognition and Language Modelling (PRLM) [3], four phonotactic based on Phone Recognisers followed by support vector machine modelling (PRSVM) [4] and an i-vector [10] based language recognition system similar to the one in [5] that makes use of single mixture Gaussian distributions for language modelling. A primary and three contrastive systems have been submitted, which differ in the number of employed sub-systems and in the back-end strategy followed. All the submitted LR systems implement Gaussian back-ends followed by linear logistic regression fusion. The *primary* system follows a greedy search strategy to find the best combination of sub-systems per evaluation condition like in [6]. The *contrastive1* system follows the same sub-system selection approach, but applies zt-norm to the phonotactic scores. The *contrastive2* system consists of the fusion of the four PRSVM sub-systems and the i-vector sub-system. The *contrastive3* system fuses the nine sub-systems.

In next section 2 a brief description of some commonalities of the sub-systems developed (see Section 2.1) is provided, together with details of each one of the nine individual sub-systems: the PRLM-LR, the PRSVM-LR and the iVECTOR-LR sub-systems are described in sections 2.2, 2.3 and 2.4, respectively. Finally, details about the back-end and fusion and about the four submitted systems are provided in section 3.

2 LR sub-system description

2.1 Sub-system commonalities

Data Pre-processing Training data provided for the evaluation consist of two sets of clean speech (around 86 hours) and noisy speech (around 22 hours) broadcast data for each one of the 6 target languages considered in the *Plenty*-training condition: Basque, Catalan, English, Galician, Portuguese and Spanish.

The training data was pre-processed to segment long data files into a set of homogeneous reduced length speech segments. In order to generate these homogenous segments, we applied our segmentation module [7], including speech-non-speech (SNS) segmentation, background classification, channel classification, gender classification and speaker clustering. After this segmentation process, we selected for each target language 5 hours of clean speech (segments with minimum duration of 15 seconds and maximum duration of 40 seconds), and 1.5 hours of noisy speech (segments with minimum duration of 10 seconds and maximum duration of 40 seconds). Table 1 shows the amount of selected segments and the average duration per segment in seconds for each target language and type of speech. After the segmentation process, all training segments are down-sampled to 8kHz sampling rate.

On the other hand, the development and evaluation data sets consist of audio extracted from Youtube videos. In this case, during the development of the systems we experimented two alternative pre-processing strategies. First, we considered removing non-speech segments detected with our segmentation module to produce a "cleaned" version of the development data set. Second, we segmented each development file in shorter speech homogeneous segments that were independently processed to obtain several language recognition scores per file. Then, we experimented some simple strategies to generate a single score. None of these two mentioned strategies provided any observable improvement with respect to the use of the whole unprocessed test segment. Consequently, it was decided to not apply any additional pre-processing to the development and evaluation data sets, besides downsampling to 8kHz.

	clean		noisy	
	#segm	mean dur. [sec]	#segm	mean dur. [sec]
Basque	778	23.1	291	17.6
Catalan	772	23.3	299	18.1
English	736	24.5	284	19.0
Galician	783	23.0	310	17.5
Portuguese	770	23.4	299	18.1
Spanish	753	23.9	282	19.2

Table 1. Training data segmentation for each target language and speech type.

Target Language Modelling One of the particularities shared among all the developed sub-systems is that a separate target language (of the *Plenty*-training condition) model was trained for clean and noisy speech. The two target models of each language are used to obtain two language-dependent scores for each speech test segment. Consequently, for every test segment a vector of 12 scores $\mathbf{x}_i$ is produced by every individual sub-system i.

2.2 PRLM-LR sub-systems

The Phone Recognition followed by Language Modelling (PRLM) systems used for Albayzin 2012 exploit the phonotactic information extracted by four individual tokenizers: European Portuguese (*pt*), Brazilian Portuguese (*bp*), European Spanish (*es*) and American English (*en*). The key aspect of this type of system is the need for robust phonetic classifiers that generally need to be trained with word-level or phonetic level transcriptions. In this case, the tokenizers are MultiLayer Perceptrons (MLP) trained to estimate the posterior probabilities of the different phonemes for a given input speech frame (and its context). For each target language and for each tokenizer a different phonotactic *n-gram* language model is trained. During test, the phonetic sequence of a given speech signal is extracted with the phonetic classifiers and the likelihood of each target language model is evaluated.

Phonetic Tokenizers The tokenization of the speech data is done with the neural networks that are part of our hybrid Automatic Speech Recognition (ASR) system named AUDIMUS [8]. The recognisers combine four MLP outputs trained with Perceptual Linear Prediction features (PLP, 13 static + first derivative), PLP with log-RelAtive SpecTrAl speech processing features (PLP-RASTA, 13 static + first derivative), Modulation SpectroGram features (MSG, 28 static) and Advanced Font-End from ETSI features (ETSI, 13 static + first and second derivatives). A phone-loop grammar with phoneme minimum duration of three frames is used for phonetic decoding.

The language-dependent MLP networks were trained with different amounts of annotated data. For the *pt* acoustic models, 57 hours of BN down-sampled data and 58 hours of mixed fixed-telephone and mobile-telephone data were used. The *bp* models were trained with around 13 hours of BN down-sampled data. The *es* networks used 36 hours of BN down-sampled data and 21 hours of fixed-telephone data. The *en* system was trained with the HUB-4 96 and HUB-4 97 down-sampled data sets, that contain around 142 hours of TV and Radio Broadcast data.

Each MLP network is characterised by the size of its input layer that depends on the particular parametrization and the frame context size (13 for PLP, PLP-RASTA and ETSI; 15 for MSG), the number of units of the two hidden layers (500), and the size of the output layer. In this case, only monophone units are modelled, resulting in MLP networks of 41 (39 phonemes +1 silence + 1 respiration) soft-max outputs in the case of *en*, 39 for *pt* (38 phonemes + 1 silence), 40 for *bp* (39 phonemes + 1 silence) and 30 for *es* (29 phonemes + 1 silence).

Phonotactics Modelling For every phonetic tokenizer, the phonotactics of each target language for every type of speech condition (clean and noisy) is modemed with a 3-gram back-off model, that is smoothened using Witten-Bell discounting. For that purpose the SRILM toolkit has been used[1].

2.3 PRSVM-LR sub-systems

Phone Recognition followed by Support Vector Machine Modelling (PRSVM) systems used for Albayzin 2012 exploit the phonotactic information extracted by the same four tokenizers described above: *pt*, *bp*, *es* and *en*. In contrast to PRLM-LR sub-systems, a recognition lattice is generated for every processed segment, from which the posterior expected n-gram counts are computed. Then, for each target language and for each tokenizer a different phonotactic SVM language model is trained with the counts vectors. During test, vectors of n-gram counts of a given speech signal are computed from the lattices obtained with the automatic phoneme recognisers and used to evaluate each language SVM model.

Phoneme Recognisers Vectors of expected n-gram counts are obtained for each speech segment based on the recognition results of our ASR system described above. Like in PRLM sub-systems, a phone-loop grammar with phoneme minimum duration of three frames is used for lattice generation.

N-gram vector extraction and dimensionality reduction The 'lattice-tool' program from the SRILM toolkit is used to compute the expected n-gram counts (up to 3-grams) of each recognition lattice. This resulting n-gram counts vector is converted to a vector of probabilities (sum 1) and it is normalised by the square root of the average probability vector computed over the whole training data set. The high-dimensionality of the n-gram vectors motivated the use of some sort of dimensionality reduction method. In practice, we applied simple frequency selection [9] with new dimensionality of 10000 elements in the four PRSVM sub-systems (this size was experimentally verified to provide good performance).

Phonotactics Modelling For every phoneme recogniser, phonotactic relations of each training data sub-set are modelled with an L2-regularised support vector classifier using the LibLinear implementation of the libSVM tool[2]. For “clean” and “noisy” SVM language model training, only “clean” and “noisy” background (non-positive) data are used respectively.

2.4 iVECTOR-LR sub-system

Total-variability modelling [10] has rapidly emerged as one of the most powerful approaches to the problem of speaker verification. In this approach, closely

[1] `http://www-speech.sri.com/projects/srilm/`

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

related to the Joint Factor Analysis [11], the speaker and the channel variabilities of the high-dimensional GMM supervector are jointly modelled as a single low-rank total-variability space. The low-dimensionality total variability factors extracted from a given speech segment form a vector, named i-vector, which represents the speech segment in a very compact and efficient way. Thus, the total-variability modelling is used as a factor analysis based front-end extractor. In practice, since the i-vector comprises both speaker and channel variabilities, in the i-vector framework for speaker verification some sort of channel compensation or channel modelling technique usually follows the i-vector extraction process.

The success of i-vector based speaker recognition has motivated the investigation of its application to other related fields, including language recognition [5, 12]. For Albayzin 2011, we have developed an i-vector based language recognition sub-system very similar to the one in [5], where the distribution of i-vectors for each language is modelled with a single Gaussian.

Feature extraction The extracted features are shifted delta cepstra (SDC) [13] of Perceptual Linear Prediction features with log-RelAtive SpecTrAl speech processing (PLP-RASTA). First, 7 PLP-RASTA static features are obtained and mean and variance normalisation is applied in a per segment basis. Then, SDC features (with a 7-1-3-7 configuration) are computed, resulting in a feature vector of 56 components. Finally, low-energy frames detected with the alignment generated by a simple bi-Gaussian model of the log energy distribution computed for each speech segment are removed.

UBM modelling A GMM-UBM of 1024 mixtures has been trained using all the training segments of Table 1. Type of speech was not distinguished and a single UBM was trained with both clean and noisy segments. In total, the number of segments considered are 6330, which corresponds almost to 22.5 hours of speech (after the low-energy removal process of the feature extraction).

Total variability and i-vector extraction The total variability factor matrix ($\mathbf{T}$) was estimated according to [14]. The dimension of the total variability subspace was fixed to 400. Zero and first-order sufficient statistics of the training sub-sets described in Table 1 were used for training $\mathbf{T}$. 10 EM iterations were applied, in the first 7 iterations only ML estimation updates were applied, while in the last 3 EM iterations both ML and minimum divergence update were applied. The covariance matrix was not updated in any of the EM iterations.

The estimated $\mathbf{T}$ matrix is used for extraction of the total variability factors of the processing speech segments as described in [14]. Finally, the resulting factor vectors are normalised to be of unit length, which we will refer as i-vectors.

Language modelling and scoring Like in [5], all the extracted i-vectors from a data sub-set of Table 1 are used to train a single mixture Gaussian distribution with full covariance matrix shared across different training sub-sets. For a given test i-vector, each Gaussian model is evaluated and log-likelihood scores are obtained.

3 The L^{2}F submitted systems

3.1 Back-end configuration and calibration

Linear Gaussian Back-End A linear Gaussian Back-End (GBE) follows every single sub-system to transform the 12 elements score-vector $\mathbf{x}_i$ (see section 2.1) to a n elements log-likelihood vector $\mathbf{s}_i$, where n equals the number of target languages in closed evaluation conditionS and equals the number of target languages plus 1 out-of-set log-likelihood in open set conditions:

$$\mathbf{s}_i = \mathbf{A}_i \mathbf{x}_i + \mathbf{o}_i \tag{1}$$

where $\mathbf{A}_i$ is the transformation matrix for system i and $\mathbf{o}_i$ is the offset vector.

Linear logistic regression (LLR) Linear logistic regression (LLR) is used to fuse the log-likelihood outputs generated by the linear GBEs of the selected sub-systems to produce fused log-likelihoods $\mathbf{l}$:

$$\mathbf{l} = \sum_i \alpha_i \mathbf{s}_i + \mathbf{b} \tag{2}$$

where α_i is the weight for sub-system i and $\mathbf{b}$ is the language-dependent shift.

During the development of the L^{2}F systems, the GBEs and the LLR fusion parameters were trained and evaluated with the development data set using a sort of 2-fold cross-validation [6]: development data is randomly split in two halves, one for parameter estimation and the other for assessment. This process is repeated using 10 different random partitions and the mean and variance of the systems' performance can be computed. For the final submission, no partition of the data was made and all the development data was used to simultaneously calibrate the GBEs and the LLR fusion. Different GBE and LLR fusion parameters have been trained for each one of the four evaluation conditions. Calibration was carried out using the FoCal Multi-class Toolkit[3].

3.2 Primary System (primary)

The L^{2}F *primary* system consists of multi-class fusion of a selected set of sub-systems. For a given test segment, the outcome of the fusion is a likelihood vector $\mathbf{l}$ of n-elements, one for each target language (plus 1 for the out-of-set in the open-set condition). The selection of sub-systems is done following an

[3] `http://niko.brummer.googlepages.com/focalmulticlass`

incremental search process using the development data. First, it is found the best single sub-system $[i]$, then the best combination of 2 sub-systems $[i, j]$ with sub-system i, then the best combination of three sub-systems with the best pair previously found, and so on. Finally, the combination of k sub-systems with the lowest minimum performance cost is selected. The search process, and consequently the selection of sub-systems, was done for each evaluation condition independently. Table 2 shows the selected sub-systems of the *primary* system for each evaluation condition. The minimum number of selected sub-systems is 4 for the PC condition and the maximum is 6 for the *Empty* training conditions. In this case, the sub-systems selected for the PC condition are always present in the other conditions. Notice, however, that there is not any restriction to make this happen and, moreover, that the order of selection may not be the same for all the conditions. An interesting observation is that PRLM and PRSVM sub-systems based on the same phonetic classifier are sometimes selected before than other phonotactic systems exploiting different phonetic recognisers. This observation may suggest that there may be some residual complementary information in n-gram and expected counts based phonotactic approaches.

3.3 First Contrastive System (contrastive1)

The L^{2}F *contrastive1* system follows the same sub-system search approach than the *primary* system with a slightly different back-end configuration. Concretely, zt-norm score normalisation is applied to each sub-system before the application of the GBE. In practice, we observed a generalised improvement of the individual sub-systems using score normalisation, with the exception of the iVECTOR-LR sub-system. Consequently, the *contrastive1* system back-end configuration applies zt-norm only to the phonotactic-based sub-systems. Table 2 details the set of sub-systems that form the *contrastive1* system per evaluation condition. In this case, the minimum number of selected sub-systems is 5. Moreover, all the sub-systems selected in the PC condition are not present in all the other conditions, in contrast to the *primary* system. Again, some phonotactic n-gram and expected counts based sub-systems using the same phonetic decoder are selected.

3.4 Second Contrastive System (contrastive2)

The L^{2}F *contrastive2* system consists of the fusion of a fixed set-up of sub-systems for all the evaluation conditions, which are the four PPRSVM-LR sub-systems plus the iVECTOR-LR one. This submission is very similar to the L^{2}F system submitted to NIST LRE2011 [15] (the NIST system incorporates an additional Gaussian supervector based sub-system [16]). Score normalisation is not applied to any of the sub-systems that form the *contrastive2* submission.

3.5 Third Contrastive System (contrastive3)

The L^{2}F *contrastive3* system is the result of the fusion of the nine developed sub-systems. No score normalisation is applied.

System	Cond.	Selected sub-systems
Primary	PC	PRLM-es, PRSVM-bp, PRSVM-es, iVECTOR
	PO	PRLM-es, PRSVM-bp, PRSVM-en, PRSVM-es, iVECTOR
	EC	PRLM-en, PRLM-es, PRSVM-bp, PRSVM-en, PRSVM-es, iVECTOR
	EO	PRLM-es, PRSVM-bp, PRSVM-en, PRSVM-es, PRSVM-pt, iVECTOR
Contrastive1	PC	PRLM-bp, PRLM-es, PRSVM-bp, PRSVM-es, iVECTOR
	PO	PRLM-es, PRSVM-bp, PRSVM-en, PRSVM-es, PRSVM-pt, iVECTOR
	EC	PRLM-bp, PRLM-en, PRLM-es, PRSVM-bp, PRSVM-es, iVECTOR
	EO	PRLM-en, PRLM-es, PRLM-pt, PRSVM-bp, PRSVM-es, iVECTOR

Table 2. Sub-systems selection in the *primary* and *contrastive1* submission for each evaluation condition. In the case of the *contrastive1* system, zt-norm is applied to the phonotactic sub-systems.

References

1. Rodríguez-Fuentes, L. J., Brümmer, N., Penagarikano, M., Varona, A., Diez, M., Bordel, G.: The Albayzin 2012 Language Recognition Evaluation Plan (Albayzin 2012 LRE). URL: `http://http://iberspeech2012.ii.uam.es/images/PDFs/albayzin_lre12_evalplan_v1.3_springer.pdf` (2012)
2. Rodríguez-Fuentes, L. J., et al.: Multi-site heterogeneous system fusions for the Albayzin 2010 language recognition evaluation. IEEE 2011 Automatic Speech Recognition and Understanding Workshop (ASRU) (2011)
3. Zissman, M.: Comparison of four approaches to automatic language identification of telephone speech. IEEE Transactions on Speech and Audio Processing, vol. 4(1), pp. 31-44 (1996)
4. Li, H., Ma, B., Lee, C.-H.: A vector space modeling approach to spoken language identification. IEEE Transactions on ASLP, vol. 15, no. 1, pp. 271–284 (2007)
5. Martínez, D., Plchot, O., Burget, L., Glembek, O., Matejka, P.: Language Recognition in iVectors Space. in Proc. Interspeech 2011, Firenze, Italy (2011)
6. Rodríguez-Fuentes, L. J., et al.: The BLZ Submission to the NIST 2011 LRE: Data Collection, System Development and Performance. in Proc. Interspeech 2012, Portland, US (2012)
7. Meinedo, H., Neto, J.: Audio Segmentation, Classification and Clustering in a Broadcast News Task, in Proc. ICASSP 2003, Hong Kong (2003)
8. Meinedo, H., Abad, A., Pellegrini, T., Trancoso, I., Neto, J: The L2F Broadcast News Speech Recognition System. in Proc. Fala2010, Vigo, Spain (2010)
9. Tong, R., Ma, B., Li, H., Chang, E. S.: Selecting phonotactic features for language recognition. in Proc. Interspeech 2010, pp. 737-740 (2010)
10. Dehak, N., Dehak, R., Kenny, P., Brummer, N., Ouellet, P., Dumouchel, P.: Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification. in Proc. Interspeech 2009, pp. 1559-1562 (2009)
11. Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P.: Joint factor analysis versus eigenchannels in speaker recognition. IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 4, pp. 1435-1447 (2007)
12. Dehak, N., Torres-Carrasquillo, P.A., Reynolds, D., Dehak, R.: Language Recognition via Ivectors and Dimensionality Reduction, in Proc. Interspeech 2011, Firenze, Italy (2011)

13. Torres-Carrasquillo, P. A. et al.: Approaches to Language Identification using Gaussian Mixture Models and Shifted Delta Cepstral Features. in Proceedings of ICSLP 2002, pp. 89-92, Denver, Colorado (2002)
14. Kenny, P., Ouellet, P., Dehak, N., Gupta, V., Dumouchel, P., A Study of Inter-Speaker Variability in Speaker Verification. IEEE Transactions on Audio, Speech and Language Processing, 16(5), pp. 980-988 (2008)
15. Abad, A.: The L^{2}F Language Recognition System for NIST LRE 2011. In The 2011 NIST Language Recognition evaluation (LRE11) Workshop, Atlanta, US (2011)
16. Campbell, W. M., Sturim, D. E., Reynolds, D. A.: Support vector machines using GMM supervectors for speaker verification. IEEE Signal Processing Letters, vol. 13(5), pp. 308-311 (2006)