University of the Basque Country + Ikerlan System for NIST 2009 Language Recognition Evaluation

M. Penagarikano[1], A. Varona[1], M. Zamalloa[2], L. J. Rodriguez[1], G. Bordel[1], J. P. Uribe[2]

(1) Department of Electricity and Electronics, University of the Basque Country, Spain
(2) Ikerlan - Technological Research Center, Spain
E-mail: mikel.penagarikano@ehu.es

1. Introduction

This paper briefly describes the language recognition system developed by the Sofware Technology Working Group (http://gtts.ehu.es) at the University of the Basque Country in collaboration with IKERLAN Technological Research Center, and submitted to the NIST 2009 Language Recognition Evaluation. The system consists of a hierarchical fusion of individual subsystems: two acoustic GLDS-SVM systems using 19 MFCC and 7-2-3-7 SDC-MFCC, two acoustic GMM-SVM systems using 19 MFCC and 7-2-3-7 SDC-MFCC, and eleven Phone-SVM systems based in 10 phonetic decoders plus a decoder-pairs phoneme co-occurrence system. Application independent well-calibrated log-likelihood-ratio scores and Bayes thresholds are used for decision making.[1]

2. Database setup

2.1. Language definition

Based on previous LRE2007 and newer VOA training data, a set of 64 languages/dialects is defined (see Table 1). Each of them is mapped either to a target LRE2009 language or to Out Of Set (OOS). For example, *Mainland* and *Taiwan* from LRE2007 and *mand* from VOA are mapped to *Mandarin*, whereas *Arabic* is mapped to *OOS*. For any input utterance, each trained language model generates a log likelihood (resulting 64 log likelihoods), which are mapped to 24 log likelihoods (corresponding to 23 target languages plus *OOS*). When discriminative models are trained (SVMs in the case of the submitted systems), only signals from languages that map to a different target language are used as negative observations. Thus, *Mainland* and *mand* signals are not used as negative samples when *Taiwan* is trained, and none of the *OOS* signals are used as negative samples when *Arabic* is trained.

[1]This work has been partially funded by the Basque Government, under program SAIOTEK, project S-PE07UN43, and the University of the Basque Country, under project EHU06/96.

Source	Languages
LRE 2007	Arabic Bengali Cantonese English-American English-Indian Farsi French German Hindi Japanese Korean Mainland Min Russian Spanish-Caribbean Spanish-Mexican Spanish-NonCaribbean Taiwan Tamil Thai Urdu Vietnamese Wu
VOA	alba amha azer bang bosn burm cant creo croa dari fren geor gree haus hind indo khme knkr kore kurd mace mand ndeb orom pash pers port russ serb shon soma span swah tibe tigr ttam turk ukra urdu uzbe viet

Table 1: Case sensitive names of trained languages. A set of 64 languages/dialects is trained, and each of them is mapped either to a target LRE2009 language or to Out Of Set (OOS).

2.2. Database sets

For each of the 64 languages, train, development and test sets are created. Development and test utterances are 30 seconds long, whereas train utterances are typically longer. Utterances from LRE2007-eval are included in the test set.

2.3. VOA data extraction

Only those languages containing enough training data are considered for development. The only exception is *engl* (English for Africa), which is discarded due to its ambiguous mapping (*EnglishAmerican* vs. *OOS*). Available data sources and extraction criteria are summarized in Table 2.

Development and test sets are populated with randomly extracted 30 second segments, using no more than 2 segments per file, and a minimum of 150 devellopment segments and 75 test segments per language. For training, the longest segment out of each file is used, with a minimum of 225 segments per language. The number of extracted segments per file is relaxed (augmented) for those languages with few utterances per set.

Source	Description	Use
Annotations	Supervised language labels, but few segments (150-200) and languages (13).	Used whenever it is possible for development (calibration), instead of training.
VOA2 labels	Lots of files, probably containing more than one program (so, probably more than one language).	Only the central third part of the signal and the most probable language label are used.
VOA3 labels	Lots of files with assigned broadcast language.	Use full signals.

Table 2: VOA2/VOA3 data extraction criteria. Supervised annotations are reserved for calibration, and only the central third of VOA2 files is used.

3. Primary System

The primary system consists of a hierarchical fusion of 15 individual subsystems: two acoustic GLDS-SVM systems using 19 MFCC and 7-2-3-7 SDC-MFCC, two acoustic GMM-SVM systems using 19 MFCC and 7-2-3-7 SDC-MFCC, and eleven Phone-SVM systems based on 10 recognizers plus a decoder-pairs phoneme co-occurrence system. All the subsystems are based on Support Vector Machines (SVM) [1], and have been developed using SVMTorch [2] and libSVM [3], the first one for dense vectors and the second one for sparse vectors.

3.1. GLDS-SVM subsystems

Two SVM-based acoustic systems are build using: (1) a parameterization based on 19MFCC plus first order deltas and (2) another one composed of 7-2-3-7 SDC based on MFCCs, both obtained with the Sautrela toolkit [4]. A polynomial expansion of degree three [5] and a Generalized Linear Discriminant Sequence kernel [6] are applied.

3.2. GMM-SVM subsystems

GMM-SVM systems use a SVM classifier on the vector space defined by the GMM parameters. The GMM of a target language is constructed by MAP adapting the means of a gender-independent UBM (Universal Background Model) consisting of 512 mixture components. The adapted mixture components means are stacked to construct the GMM supervectors. Based on the previously mentioned parameterizations (MFCC and SDC), two GMM-SVM subsystems are implemented. UBM and GMMs are estimated using Sautrela.

3.3. Phone-SVM subsystems

Audio files are filtered using a Voice Activity Detector (VAD) to get speech segments. Speech segments are decoded using ten open-loop phone decoders for seven languages:

- Three of the phone decoders (for Spanish, English and Basque) were trained using clean laboratory-condition speech originally recorded at 16kHz and downsampled to 8kHz. The phone decoder for Spanish was trained using the Albayzin database [7], whose training corpus contains about 4,3 hours of speech. The phone decoder for English was trained using the Wall Street Journal acoustic database, whose training corpus amounts to about 20 hours of speech. The phone decoder for Basque was trained using about 2,6 hours of speech.

- The remaining seven decoders were trained using telephone-quality speech. For Spanish, the Dihana database [8] was used, which contains about 5,1 hours of speech. For Basque, the training corpus, consisting of read sentences and containing thousands of speakers, was about 50 hours long. The remaining five decoders were trained on the CSLU Multilanguage Telephone Corpus, using five phonetically labeled languages: English (about 13 hours), German (about 3 hours), Hindi (about 2,7 hours), Japanese (about 2,5 hours) and Mandarin (about 2,8 hours). The training process consisted of four steps: (1) the set of labeled utterances is used to bootstrap phone models; (2) then, unlabeled utterances are automatically transcribed based on those initial phone models; (3) phone models are reestimated based on the whole database, using both manual and automatically obtained phonetic labels; and (4) step (3) is repeated until a maximum number of iterations is reached.

Phone decoders are built using HTK. Acoustic parameters consist of 10 Mel Frequency Cepstral Coefficients (MFCCs), energy and their first and second derivatives, calculated on 25ms windows, with a window step of 10ms. Phone models are defined as context-independent, 3-state, left to right continuous Hidden Markov Models (HMM) with a mixture of 16 Gaussians per state. Decoders used in this work involve between 22 and 40 phones.

Each target language is modeled by means of phone statistics produced by phone decoders. In particular, 1-best phone sequences are used to estimate 1-grams, 2-grams and 3-grams which are stored in a single vector. An SVM is estimated for each pair (target language, phone decoder).

The approach described above use phone n-grams to feed an SVM-based classifier. Phone n-grams are computed independently for each decoder, producing 10 diffe-

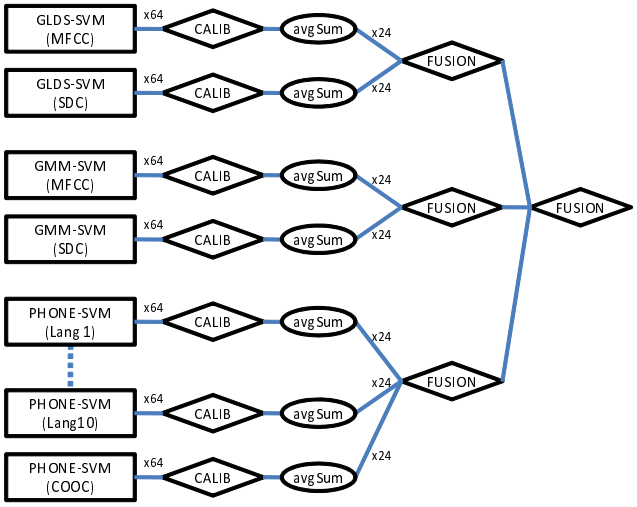

Figure 1: 64 language probabilities are calibrated and mapped to 24 target language probabilities. Then, a hierarchical fusion is performed.

rent streams of information. This way time-synchronous relations between different phone sequences are not taken into account. However, phone decoders could be merged and the statistics feeding the SVM counted in two axes: decoder-internal and across decoders. This way, decodings would be merged and then an SVM applied to the resulting lattice. We considered this too complex for a first approach, so we simply count phone co-occurrences for each pair of phone decoders on a frame-by-frame basis. Let consider an input utterance X consisting of T frames, and the optimal decodings produced by decoders A and B: $P_A(X) = \{p_A(t),\ t \in [1,T]\}$ and $P_B(X) = \{p_B(t),\ t \in [1,T]\}$, where $p_A(t)$ and $p_B(t)$ are the phones at frame t in the optimal decodings by decoders A and B, respectively. Then the count $c(a,b)$ is computed as follows:

$$c(a,b) = \sum_{t=1}^{T} \delta(p_A(t) = a)\delta(p_B(t) = b)$$

These counts are computed for each pair of decoders, which amounts to $C = \sum_{j=1}^{D-1} N_j \cdot \sum_{k=j+1}^{D} N_k$ counts, where D is the number of decoders and N_j the number of phones in decoder j. In this work, using 10 decoders, C=44169. These counts are stored in a single vector which feeds an SVM-based classifier.

3.4. Fusion

Linear logistic regression fusion and calibration is performed using the FoCal Toolkit [9] on the development dataset. For each subsystem, the 64 log-probabilities are calibrated before mapping them to 24 log-probabilities (corresponding to 23 target languages plus *OOS*) according to

Primary System						
	GLDS-SVM		GMM-SVM		Phone-SVM	
	MFCC	SDC	MFCC	SDC	Phono	Co-occ
Param	3,02	0,89	(done)	(done)	0,53	(done)
pre-SVM	2,17	4,06	12,50	14,71	7,35	(done)
SVM	8,00	25,50	3,55	5,25	6,50	4,50
Total	13,19	30,45	16,05	19,96	14,38	4,50
Speed	0,06	0,139	0,073	0,091	0,066	0,021
Total	**98,53**					
Speed	**0,45**					

Table 3: Primary system processing time (in hours) and speed (in xRT).

the folowing average sum:

$$log\left(P\left(X\,|T_i\right)\right) = log\left(\frac{1}{|S_i|}\sum_{L\in S_i} P\left(X\,|L\right)\right)$$

where S_i is the set of languages that map to target language T_i, and X is an input utterance. Then, a two-step hierarchical fusion is performed with all the 15 subsystems, as it is shown in Figure 1.

3.5. Processing speed

The NIST LRE 2009 evaluation data is about 218 hours long. Table 3 shows the time employed at each processing stage. The step labeled as *pre-SVM* contains the time needed for GLDS kernel expansion, GMM MAP calculation and phone decoding. Experiments where carried out on a dual Opteron 2378 server (8 cores) with 16 GBytes of memory. Note that, when possible, the processing was parallelized in 8 threads.

3.6. System performance

Figures 2a and 2b show the DET curves for the primary system in *Closed-Set* and *Open-Set* conditions, computed over the development and test sets.

4. Contrastive System

The submitted contrastive system uses both the development ant test sets for calibration and fusion. Figures 2c and 2d show the DET curves for the contrastive system in *Closed-Set* and *Open-Set* conditions, computed over the union of the development and test sets.

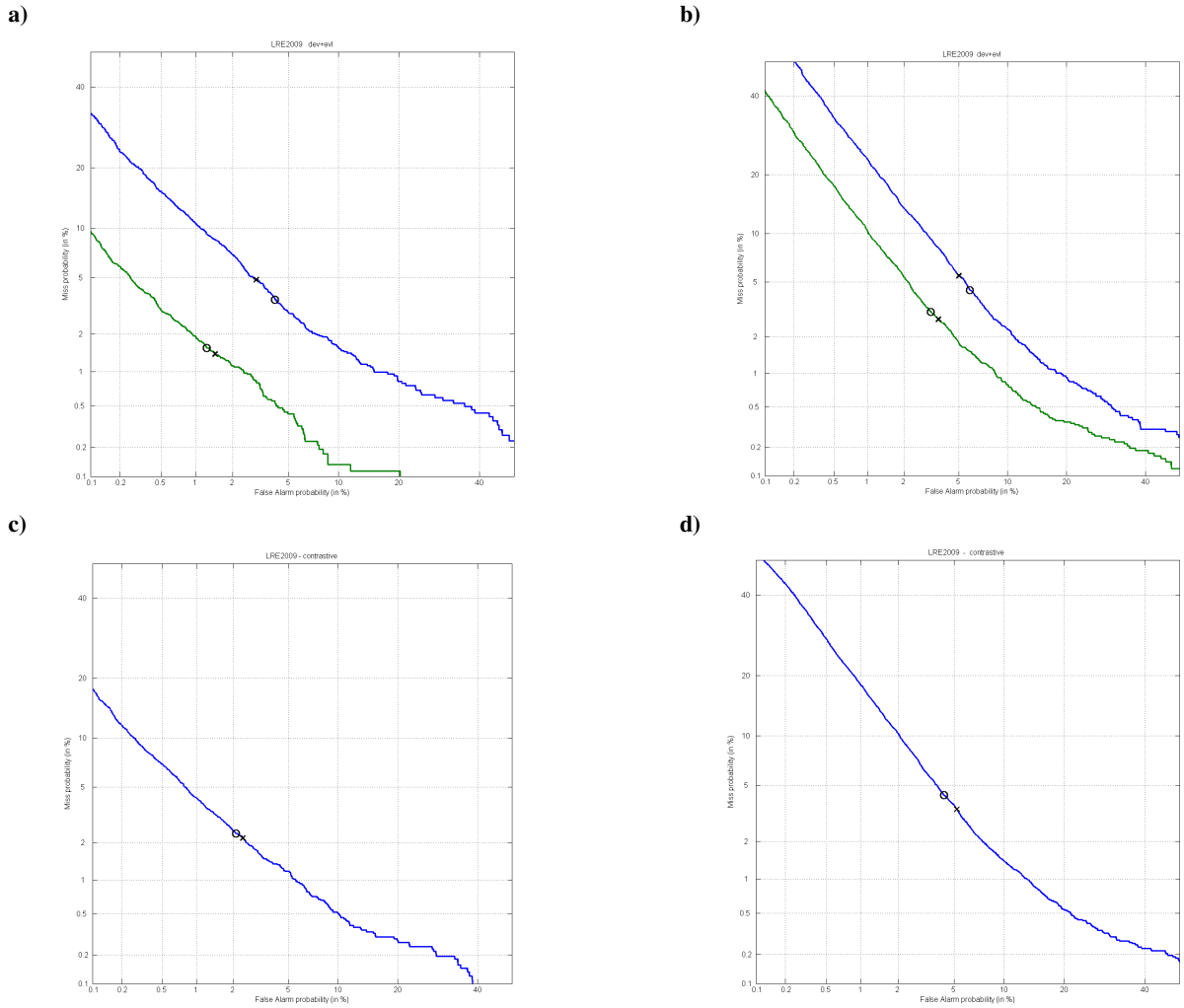

Figure 2: DET curves of **a)** Primary *Closed-Set* **b)** Primary *Open-Set* **c)** Contrastive *Closed-Set* and **d)** Contrastive *Open-Set*.

5. References

[1] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," Computer Speech and Language, vol. 20, no. 2-3, pp. 210–229, 2006.

[2] Collobert, R. and Bengio, S. 2001. SVMTorch: support vector machines for large-scale regression problems. J. Mach. Learn. Res. 1 (Sep. 2001), 143-160.

[3] Chang, C. C.; Lin, C. J. "LIBSVM: a library for support vector machines", 2001.

[4] M. Penagarikano and G. Bordel, "Sautrela: A Highly Modular Open Source Speech Recognition Framework", in Proceedings of the ASRU Workshop, 2005.

[5] W. Wan and W. Campbell, "Support vector machines for speaker verification and identification," in Proc. of IEEE International Workshop on Neural Networks for Signal Processing, 2000, pp. 775–784

[6] W. Campbell, "Generalized linear discriminant sequence kernels for speaker recognition," in Proc. of ICASSP, 2002, pp. 161–164.

[7] A. Moreno, D. Poch, A. Bonafonte, E. Lleida, J. Llisterri, J.B. Mariño, and C. Nadeu, "ALBAYZIN Speech Database: Design of the Phonetic Corpus", in Eurospeech'93. Berlin, Germany, 21-23 September 1993. Vol. 1 pp. 175-178.

[8] J.M. Benedí E. Lleida, A. Varona, M.J. Castro, I. Galiano, R. Justo, I. Lopez, A. Miguel. Design and acquisition of a telephone spontaneous speech dialogue corpus in Spanish: DIHANA. LREC2006, Italia 2006.

[9] N. Brümmer et al. "Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006." IEEE Transactions on Audio, Speech and Signal Processing, 15(7) pp. 2072-2084, 2007.