Improved Modeling of Cross-Decoder Phone Co-occurrences in SVM-based Phonotactic Language Recognition

Mikel Penagarikano, Amparo Varona, Luis Javier Rodríguez-Fuentes, Germán Bordel

GTTS, Department of Electricity and Electronics
University of the Basque Country, Spain
mikel.penagarikano@ehu.es

Abstract

Most common approaches to phonotactic language recognition deal with several independent phone decodings. These decodings are processed and scored in a fully uncoupled way, their time alignment (and the information that may be extracted from it) being completely lost. Recently, a new approach to phonotactic language recognition has been presented [1], which takes into account time alignment information, by considering cross-decoder phone co-occurrences at the frame level, under two language modeling paradigms: smoothed n-grams and Support Vector Machines (SVM). Experiments on the NIST LRE2007 database demonstrated that using phone co-occurrence statistics could improve the performance of baseline phonotactic recognizers. In this paper, two variants of the cross-decoder phone co-occurrence SVM-based approach are proposed, by considering: (1) n-grams (up to 3-grams) of phone co-occurrences; and (2) co-occurrences of phone n-grams (up to 3-grams). To evaluate these approaches, a choice of open software (Brno University of Technology phone decoders, LIBLINEAR and *FoCal*) was used, and experiments were carried out on the NIST LRE2007 database. Unlike those presented in [1], the two approaches presented in this paper outperformed the baseline phonotactic system, yielding around 16% relative improvement in terms of EER. The best fused system attained a 1,88% EER (a 30% improvement with regard to the baseline system), which supports the use of cross-decoder dependencies for language modeling.

1. Introduction

Phonotactic language recognizers exploit the ability of phone decoders to convert a speech utterance into a sequence of phones containing acoustic, phonetic and phonological information. Models for target languages are built by decoding hundreds or even thousands of training utterances and using the phone-sequence (or phone-lattice) statistics (typically, counts of n-grams) in different ways. Since training data feature a wide range of speakers and diverse linguistic contents, being *language* the common factor, it is expected that phone statistics reflect language-specific characteristics.

This work has been supported by the Government of the Basque Country, under program SAIOTEK (project S-PE09UN47), and the Spanish MICINN, under Plan Nacional de I+D+i (project TIN2009-07446, partially financed by FEDER funds).

The most common phonotactic approaches are the so called PPRLM (Parallel Phone Recognizers followed by Language Models) [2], referred to as Phone-LM in this paper, and Phone-SVM (Support Vector Machines applied on counts of phone n-grams) [3]. In both cases, N phone decoders are applied to the input utterance, yielding N phone decodings (or lattices). The output of the phone decoder i ($i \in [1, N]$) is scored for each target language j ($j \in [1, L]$), by applying the model $\lambda(i, j)$ (estimated using the outputs of the phone decoder i for the training database, taking j as the target language). Scores for the subsystem i are calibrated, typically by means of a Gaussian backend. Sometimes, a t-norm [4] is applied before calibration. Finally, $N \times L$ calibrated scores are fused applying linear logistic regression, to get L final scores for which a minimum expected cost Bayes decision is taken, according to application-dependent language priors and costs (see [5, 6] for details). Figure 1 shows the structure of a typical phonotactic language recognizer.

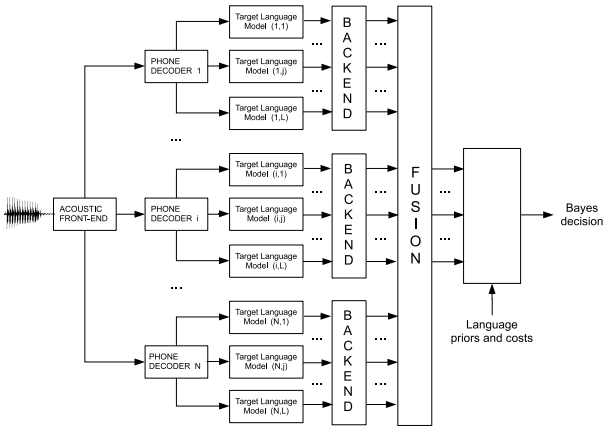

Figure 1: A phonotactic language recognition system.

However, the above described structure defines N independent data processing channels, and no cross-decoder dependencies are exploited for language modeling, information being fused only at the score level. The idea of using phonetic information in the cross-stream (cross-decoder) dimension was first applied for speaker recognition in the Johns Hopkins University (JHU) 2002 Workshop [7], where two decoupled time and cross-stream dimensions were modelled separately and integrated at the score level. Some years later, cross-stream dependencies were also used via multi-string alignments in a language recognition application [8].

Recently, a simple approach has been proposed which takes into account cross-decoder phone co-occurrences at the frame level [1]. In that approach, phone segmentation is extracted as side information from 1-best phone decodings, and allows us to consider the *co-occurrence* of N phone labels (one per decoder) at each frame. This way, a frame-synchronous sequence of multi-phone labels can be defined and used for modeling purposes, following either the Phone-LM or the Phone-SVM approaches. The simplest case consists of considering just two decoders A and B (out of N) and using sequences of two-phone labels, which can be processed and modelled exactly the same way as single-phone sequences (see Figure 2).

In fact, $N(N-1)/2$ of such 2-decoder subsystems can be defined and fused at the score level to get a full 2-phone co-occurrence system. This configuration can be easily generalized to k-decoder subsystems ($k = 3, 4, \ldots, N$). As for n-grams, the number of possible k-phone co-occurrences increases exponentially with k, so in this work only 2-phone and 3-phone co-occurrences will be considered.

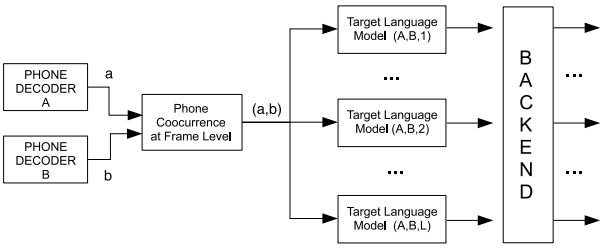

Figure 2: A 2-decoder phone co-occurrence language recognition subsystem.

In experiments on the NIST LRE2007 database, using Brno University of Technology (BUT) decoders for Czech, Hungarian and Russian [9], it was shown that fusing baseline phonotactic systems with systems based on cross-decoder phone co-occurrences led to improved performance in all the cases (see [1] for details). However, systems based on cross-decoder phone co-occurrences did not outperform the baseline phonotactic systems. On the other hand, systems using 2-phone co-occurrences yielded better performance than those using 3-phone co-occurrences. When using 2-phone co-occurrences, the Phone-LM approach outperformed Phone-SVM, probably due to the fact that only unigram statistics were used in Phone-SVM, whereas up to 4-grams were considered in Phone-LM.

The work presented in this paper focuses on exploring different ways of exploiting the information contained in 2-phone and 3-phone co-occurrence sequences in SVM-based phonotactic language recognition. Two variants of the approach presented in [1] are proposed. In the first one, SVM vectors consist of counts of up to 3-grams (instead of just unigrams) of 2-phone and 3-phone co-occurrences. The second one does not consider n-grams of phone co-occurrences, but co-occurrences of phone n-grams (up to 3-grams). These approaches have been evaluated using open software (BUT phone decoders, LIBLINEAR and *FoCal*) and a relevant database (NIST LRE2007).

The rest of the paper is organized as follows. The baseline phonotactic system used in this work is described in Section 2. Approaches based on cross-decoder phone co-occurrences are described in Section 3. The experimental setup is briefly described in Section 4. Results of language recognition experiments on the NIST LRE2007 database (pooled for all the target languages) are presented and discussed in Section 5. Finally, conclusions and potential lines for future work are outlined in Section 6.

2. Baseline SVM-based Phonotactic Language Recognizer

The TRAPS/NN phone decoders developed by the Brno University of Technology (BUT) for Czech (CZ), Hungarian (HU) and Russian (RU) [9] are the core elements of all the systems developed in this work. BUT decoders have been previously used by other groups (besides BUT [10], the MIT Lincoln Laboratory [11]) as the core elements of their phonotactic language recognizers, with high-accuracy results. Non phonetic units appearing in the decodings (*int*, *pau* and *spk*) are mapped to silence (*sil*). After this, output dimensions for BUT decoders are 43 (CZ), 59 (HU) and 49 (RU), respectively. Before doing phone tokenization, an energy-based voice activity detector is applied to split and remove non-speech segments from the signals. Since each BUT decoder runs an acoustic front-end, it can be seen as a black box which takes a speech signal as input and gives the 1-best phone decoding as output. Regarding channel compensation, noise reduction, etc. all the systems presented in this paper rely on the acoustic front-end embedded in BUT decoders.

In the baseline system, phone sequences are modelled by means of Support Vector Machines (SVM). SVM vectors consist of counts of phone n-grams (up to trigrams), weighted as proposed in [12]. A Crammer and Singer

solver for multiclass SVMs with linear kernels has been applied, by means of LIBLINEAR [13] (much faster than libSVM [14] when using linear kernels), which has been modified by adding some lines of code to compute regression values.

Finally, the baseline system is built by fusing the scores of three calibrated SVM-based phonotactic subsystems, for Czech, Hungarian and Russian decoders. The *FoCal* toolkit is used for calibration and fusion [5, 6].

3. Improved Modeling of Cross-Decoder Phone Co-occurrences

In the following paragraphs, we describe two approaches that make use of cross-decoder co-occurrences to model target languages in SVM-based phonotactic language recognition. The first approach uses n-grams of cross-decoder phone co-occurrences; the second one, counts of cross-decoder co-occurrences of phone n-grams.

3.1. Approach 1: n-grams of phone co-occurrences

Let us consider an input sequence of feature vectors $X = (X_1, \ldots, X_T)$, T being the length of X, and assume that N phone decoders are available. The 1-best phone segmentations produced by such decoders are given by: $S^{(d)}(X) = \{s_1^{(d)}, \ldots, s_T^{(d)}\}$, $d \in [1, N]$, $s_t^{(d)}$ being the phone label produced by decoder d at frame t. A cross-decoder time-synchronous (frame level) k-phone co-occurrence is defined by the k-tuple $c^\pi(t) = (s_t^{(d_1)}, s_t^{(d_2)}, \ldots, s_t^{(d_k)})$, $\pi = (d_1, d_2, \ldots, d_k)$ being a choice of k decoders, with $k \in [2, N]$. A sequence of 3-phone co-occurrences (corresponding to 3 decoders) is depicted in Figure 3. Note that a sequence of k-phone co-occurrences $C^\pi = (c^\pi(1), c^\pi(2), \ldots, c^\pi(T))$ includes information from both time and cross-stream dimensions.

We make the assumption that sequences of k-phone co-occurrences are somehow language-specific. So, a language recognition system could be built by counting such events for a training database and estimating SVM-based language models, which should be able to discriminate target languages from each other. There can be defined $N!/k!(N-k)!$ of such systems, which could be applied on an independent way and their scores fused to get a full cross-decoder phone co-occurrence language recognition system. To keep computational costs reasonably low, in this work frame-level phone co-occurrences are considered only for $k = 2$ and $k = 3$ decoders.

In this work, we aimed to model cross-decoder segmental (phone-level) dependencies, not cross-decoder frame-level dependencies. The use of frame-level phone labels was motivated just by the need to synchronize phone decodings with each other. A sort of segmental representation can be recovered by reducing each sequence of repeated co-occurrences to a single label. However, when analyzing frame-level sequences, two types of segments can be identified: (1) *stationary segments*, corresponding to relatively long portions of speech for which decoders keep the same labels; and (2) *transitional segments*, appearing at phone borders, resulting from the fact that each decoder detects phone transitions at different points (see an example in Figure 3). We hypothesize that phone co-occurrences corresponding to transitional segments reflect random variations in the way each decoder determines phone boundaries and may distort language models. So, before reducing long sequences (stationary segments), short sequences (transitional segments) are filtered out. In this work, this is done by replacing the co-occurrence label at each frame by the mode computed on a window of size 7 around it (applied iteratively until convergence) which roughly makes sequences of length shorter than 3 to be *absorbed* by the surrounding sequences (see an example in Figure 3).

The resulting sequences of phone co-occurrences are then used to compute n-grams, which can be applied either to estimate SVM parameters or to score an input signal with regard to SVM-based language models. In [1], a complete representation of phone co-occurrences was used, so that SVM vectors comprised between 2000 and 3000 unigrams for 2-decoder configurations and more than 124000 unigrams for a 3-decoder configuration. Under such a complete representation, including bigrams and trigrams of phone co-occurrences in SVM vectors was prohibitive. In this work, a sparse representation is used instead, which involves only the n-grams seen more than 30 times in training data. This way, the representation is bounded above by the amount of data used to compute the statistics. In practice, the size of SVM vectors defined this way (including up to trigrams) is always less than 10000.

3.2. Approach 2: co-occurrences of phone n-grams

The second approach consists of considering cross-decoder co-occurrences of phone n-grams, generalizing the first approach, which is limited to phone unigrams. This generalization involves an important change when counting co-occurrences at frame level: for any given decoder, up to n n-grams can overlap at each frame t, which means that up to n^k phone n-grams can co-occur at the same frame for a choice of k decoders. So, a procedure must be designed for distributing co-occurrence counts at frame level. This procedure will allow us to circumvent the issue of lack of synchronization among decoders at phone borders. In this work, we consider only cross-decoder co-occurrences of n-grams with the same n. Though possible, mixed co-occurrences (unigrams with bigrams, bigrams with trigrams, etc.) are not considered.

Let us consider an input sequence of feature vectors $X = (X_1, \ldots, X_T)$ and a choice of k decoders $\pi = (d_1, \ldots, d_k)$. Let $\Gamma_n^{(d)}(t)$ be the set of n-grams overlapping at frame t in decoder d. Let $w_n^{(d)}(t, i)$

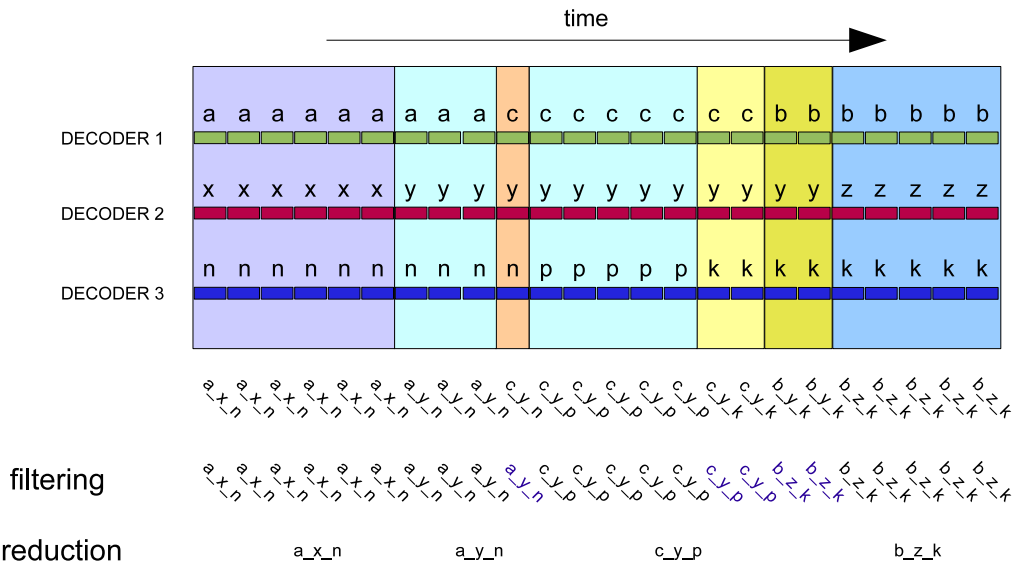

Figure 3: Approach 1 (3-decoder configuration): (1) phone co-occurrence labels are built by concatenating phone labels on a frame-by-frame basis; (2) to handle transitional segments, a mode filter is iteratively applied (until convergence) on a sliding window of 7 frames centered on the analyzed frame; and (3) repeated labels are reduced to a single label.

be one of such n-grams and $f_n^{(d)}(t,i)$ the number of frames it spans, with $i \in [1, |\Gamma_n^{(d)}(t)|]$. Note that $|\Gamma_n^{(d)}(t)| = n$ for all t except for a number of frames at the borders of X, where $1 \leq |\Gamma_n^{(d)}(t)| < n$. Let $c_n^\pi(t,\nu) = (w_n^{d_1}(t,i_1), \ldots, w_n^{d_k}(t,i_k))$ be a co-occurrence of k phone n-grams, for a choice of n-grams $\nu = (i_1, \ldots, i_k)$, with $1 \leq i_j \leq |\Gamma_n^{(d_j)}(t)|$, for $j \in [1,k]$. See Figure 4 and the related examples below to better understand these definitions.

In this approach, each decoder $d_j \in \pi$ makes its own contribution to the count of a given co-occurrence of phone n-grams at a given frame. The key concepts are: (1) each phone n-gram is counted once for each decoder, so its count is distributed among all the frames it spans; and (2) the contribution corresponding to a given phone n-gram at a given frame for a given decoder is distributed among all the combinations of phone n-grams at that frame for the remaining decoders. Taking into account these principles, we get the following expression:

$$count(c_n^\pi(t,\nu), d_j) = \frac{1}{f_n^{(d_j)}(t,i_j) \cdot \prod_{\substack{l=1 \\ l \neq j}}^{k} |\Gamma_n^{(d_l)}(t)|} \quad (1)$$

The count for $c_n^\pi(t,\nu)$ is computed as the average contribution over all the decoders:

$$count(c_n^\pi(t,\nu)) = \frac{1}{k} \sum_{j=1}^{k} count(c_n^\pi(t,\nu), d_j) \quad (2)$$

Finally, the count corresponding to a given co-occurrence of phone n-grams $b_n^\pi = (v_n^{(d_1)}, \ldots, v_n^{(d_k)})$ is computed by adding the counts for all the frames in the sequence where it appears:

$$count(b_n^\pi) = \sum_{t=1}^{T} \sum_{\forall \nu} \delta(b_n^\pi, c_n^\pi(t,\nu)) \cdot count(c_n^\pi(t,\nu)) \quad (3)$$

In practice, counts are computed in two passes. The first pass computes and stores $|\Gamma_n^{(d)}(t)|$ and $f_n^{(d)}(t,i)$ for each decoder d and each frame t. Starting from the previously stored values, the second pass accumulates the counts of phone n-grams on a frame-by-frame basis, applying equation 2 for each combination ν of phone n-grams appearing at frame t.

In this work, we consider cross-decoder co-occurrences of unigrams, bigrams and trigrams for each combination of $k = 2$ and $k = 3$ decoders (out of $N = 3$). An example for $k = 2$ decoders ($\pi = (1,2)$) including up to bigrams, is shown in Figure 4. Let us consider the shaded frame ($t = 15$) in Figure 4. The sets of n-grams appearing at that frame are:

$$\Gamma_1^{(1)}(15) = \{c\} \qquad \Gamma_2^{(1)}(15) = \{ac, cb\}$$
$$\Gamma_1^{(2)}(15) = \{y\} \qquad \Gamma_2^{(2)}(15) = \{xy, yz\}$$

and the number of frames they span:

$$f_1^{(1)}(15,1) = 8 \qquad f_2^{(1)}(15,1) = 17$$
$$f_1^{(2)}(15,1) = 13 \qquad f_2^{(1)}(15,2) = 15$$
$$f_2^{(2)}(15,1) = 19$$
$$f_2^{(2)}(15,2) = 18$$

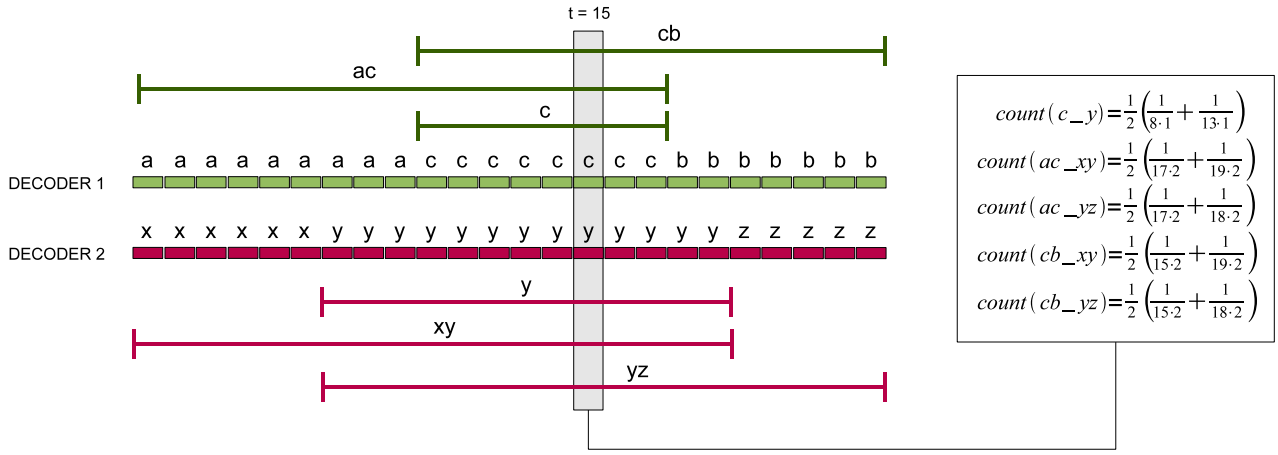

Figure 4: Approach 2 (2-decoder configuration, up to bigrams): (1) each n-gram is counted once for each decoder, so its count is distributed among all the frames it spans; (2) the contribution corresponding to a given n-gram at a given frame for a given decoder is distributed among all the combinations of n-grams appearing at that frame for the remaining decoders; and (3) the count corresponding to a given n-gram at a given frame is computed as the average contribution over all decoders.

Starting from these values and according to equation 2, the counts of co-occurrences of phone n-grams are computed as follows:

$$
\begin{aligned}
count(c\_y) &= \frac{1}{2} \cdot \left(\frac{1}{8 \cdot 1} + \frac{1}{13 \cdot 1} \right) \\
count(ac\_xy) &= \frac{1}{2} \cdot \left(\frac{1}{17 \cdot 2} + \frac{1}{19 \cdot 2} \right) \\
count(ac\_yz) &= \frac{1}{2} \cdot \left(\frac{1}{17 \cdot 2} + \frac{1}{18 \cdot 2} \right) \\
count(cb\_xy) &= \frac{1}{2} \cdot \left(\frac{1}{15 \cdot 2} + \frac{1}{19 \cdot 2} \right) \\
count(cb\_yz) &= \frac{1}{2} \cdot \left(\frac{1}{15 \cdot 2} + \frac{1}{18 \cdot 2} \right)
\end{aligned}
$$

For estimating the SVMs corresponding to target languages, counts computed this way are accumulated for a training database, SVM vectors being built with the M highest counts ($M = 100000$ in this work). Note that counts of co-occurrences of unigrams, bigrams and trigrams are put together in a single representation, which, as for the approach 1, includes information from both time (phone n-grams) and cross-stream (co-occurrence) dimensions.

For scoring purposes, given an input sample X, we first obtain 1-best decodings and segmentations, then count phone n-gram co-occurrences and use them to build an M-dimensional vector. Finally, this vector is scored with regard to SVMs. Note that counts of co-occurrences of phone n-grams not appearing among those with the M highest counts in the training database are not used for scoring.

4. Experimental Setup

4.1. Training, development and test corpora

Training and development data were limited to those distributed by NIST to all LRE2007 participants: (1) the Call-Friend Corpus; (2) the OHSU Corpus provided by NIST for LRE05; and (3) the development corpus provided by NIST for LRE07. For development purposes, 10 conversations per language were randomly selected, the remaining conversations being used for training. Each development conversation was further split in segments containing 30 seconds of speech. Evaluation was carried out on the LRE07 evaluation corpus, specifically on the 30-second, closed-set condition (primary evaluation task for the LRE07).

4.2. Evaluation measures

Most authors compare the performance of language recognition systems either globally (but not numerically) by means of Detection Error Tradeoff (DET) plots, or numerically (but not globally, and not at the optimal operation point) by means of Equal Error Rates (EER). In this work, systems will be also compared in terms of the so called C_{LLR} [15], which is used as an alternative performance measure in NIST evaluations. We internally consider C_{LLR} as the most relevant performance indicator, for two reasons: (1) C_{LLR} allows us to evaluate system performance globally by means of a single numerical value, which is somehow related to the area below the DET curve, provided that scores can be interpreted as log-likelihood ratios; and (2) C_{LLR} does not depend on application costs; instead, it depends on the calibration of scores, an important feature of detection systems.

5. Results

Table 1 shows EER and C_{LLR} performance in language recognition experiments on the LRE2007 database using the baseline phonotactic system and the cross-decoder co-occurrence approaches proposed in this work. First of all, note that we call *systems* either to those that, for a given approach, are obtained by fusing subsystems working on subsets of one or two decoders, or to those working on the whole set of three decoders. For the sake of completeness (to allow complete analyses), the performance of subsystems is also shown in Table 1, and rows corresponding to fused systems are shaded.

Table 1: Performance (EER and C_{LLR}) of the baseline phonotactic system and systems based on the cross-decoder co-occurrence approaches proposed in this work.

		EER	C_{LLR}
Baseline	CZ	5,67%	0,8259
	HU	5,10%	0,7434
	RU	5,64%	0,8016
	Fusion	2,69%	0,3981
Approach 1 (k=2)	CZ-HU	4,07%	0,5661
	CZ-RU	4,53%	0,6526
	HU-RU	3,79%	0,5109
	Fusion	2,27%	0,3393
Approach 1 (k=3)	CZ-HU-RU	4,34%	0,6500
Approach 2 (k=2)	CZ-HU	3,32%	0,4506
	CZ-RU	3,58%	0,5276
	HU-RU	2,75%	0,4140
	Fusion	2,24%	0,3223
Approach 2 (k=3)	CZ-HU-RU	3,90%	0,5724

Both approaches outperformed the baseline system when using combinations of $k = 2$ decoders. Approach 2 (2,24% EER, C_{LLR} = 0,3223) was slightly better than Approach 1 (2,27% EER, C_{LLR} = 0,3393), the relative improvement they provide being around 16% (with regard to baseline 2,69% EER). Note that the difference between Approach 1 and Approach 2 is relatively higher in terms of C_{LLR} than in terms of EER. This reflects the difference between their DET curves (see Figures 5 and 6), which is not so noticeable at the EER line.

Regarding subsystems, note that 2-decoder subsystems performed consistently better than 1-decoder subsystems, being HU-RU the combination that yielded best results. Moreover, 2-decoder subsystems based on Approach 2 performed remarkably better than 2-decoder subsystems based on Approach 1. However, those differences were not so noticeable after fusion. This may indicate that subsystems based on Approach 2 were quite redundant, or in other words, that they shared a great amount of information. This would explain why fusion did not recover for them so much information as for subsystems based on Approach 1.

A somehow unexpected result was that under 3-decoder configurations both approaches showed a poor performance compared to the baseline system (see Figures 5 and 6). We knew that robustness issues could arise from the huge amount of co-occurrences that are theoretically possible when dealing with $k \geq 3$ decoders. In Approach 1, the number of transitional segments may explode as the number of decoders increases, thus producing noisy sequences of phone co-occurrences. We tried to avoid short segments by means of a mode filter, but taking into account system performance (4,34% EER, worse than those of CZ-HU and HU-RU subsystems), it revealed insufficient. In Approach 2, a huge number of cross-decoder phone n-gram combinations could appear, specially in the case of 3-grams. To get reliable estimations of counts of co-occurrences of n-grams, a huge database would be required. This was not the case, so we limited the SVM vector to the 100000 highest counts. This way, robustness issues may be overcome but a new issue could arise: lack of coverage. Again, attending to system performance (3,90% EER, worse than those of 2-decoder subsystems for the Approach 2), we conclude that n-gram co-occurrences cannot be suitably covered with the 100000 highest counts. A lesson learned is that co-occurrence information can be effectively extracted in 2-decoder configurations (less sensitive to robustness and coverage issues) and recovered by means of fusion. Despite this, we will keep on searching for an exit to the combinatorial dead end intrinsic to cross-decoder approaches, and future work will be partly devoted to that task.

Table 2: Performance (EER and C_{LLR}) of various fused systems, involving the baseline system and systems based on Approach 1 (A1) and Approach 2 (A2).

Fused Systems	EER	C_{LLR}
A1 (k=2) + A1 (k=3)	2,21%	0,3388
A2 (k=2) + A2 (k=3)	2,28%	0,3280
Baseline + A1 (k=2)	1,92%	0,3054
Baseline + A2 (k=2)	1,88%	0,3064
Baseline + A1 (k=3)	2,38%	0,3472
Baseline + A2 (k=3)	2,15%	0,3582
Baseline + A1 (k=2) + A1 (k=3)	2,02%	0,3056
Baseline + A2 (k=2) + A2 (k=3)	1,90%	0,3158

Table 2 show the EER and C_{LLR} performance of various system fusions. Systems based on 3-decoder co-occurrences did not significantly improve the performance of systems based on 2-decoder co-occurrences. This only means that they basically model the same cross-decoder information and do not complement each other well. This argument is supported by the fact that when fused with the baseline phonotactic system, systems based on 3-decoder co-occurrences provided remarkable improvements, leading to 2,38% EER (11,52%

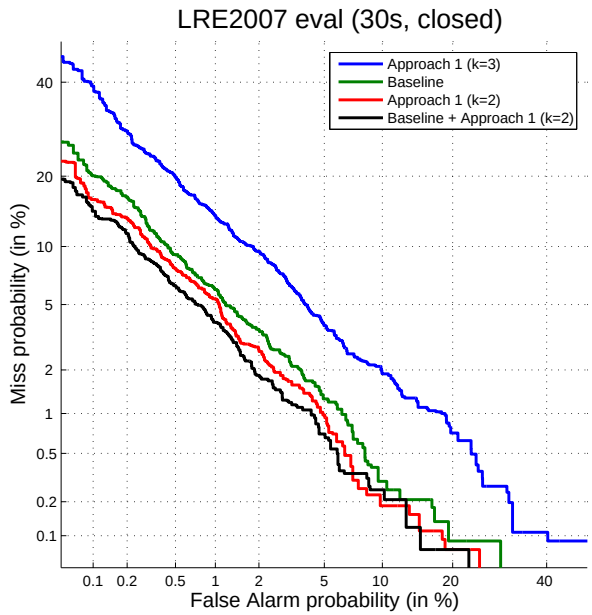

Figure 5: Pooled DET curves for the baseline phonotactic language recognition system, two systems based on Approach 1 (n-grams of cross-decoder phone co-occurrences, for $k = 2$ and $k = 3$ decoders) and the fused system Baseline + Approach 1 ($k = 2$).

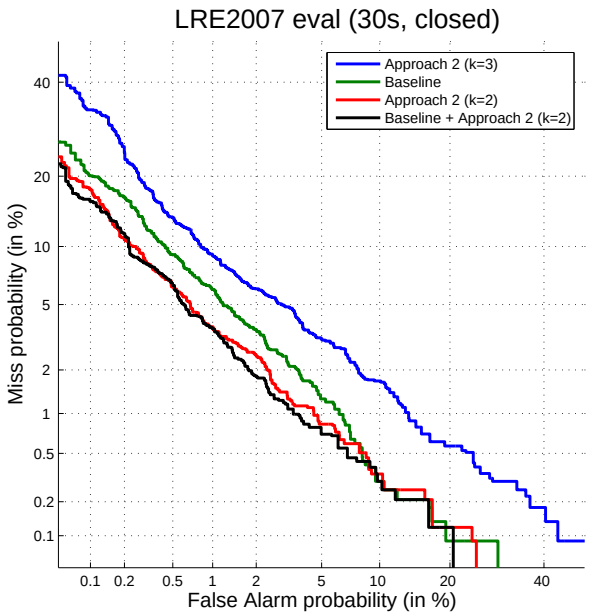

Figure 6: Pooled DET curves for the baseline phonotactic language recognition system, two systems based on Approach 2 (cross-decoder co-occurrences of phone n-grams, for $k = 2$ and $k = 3$ decoders) and the fused system Baseline + Approach 2 ($k = 2$).

relative improvement) and 2,15% EER (20,07% relative improvement) for approaches 1 and 2, respectively.

The best performance was achieved when fusing the baseline system with systems based on 2-decoder co-occurrences, which led to 1,92% EER (28,62% relative improvement) and 1,88% EER (30,11% relative improvement) for approaches 1 and 2, respectively (shaded rows in Table 2). There is, however, an important difference between approaches 1 and 2, which regards how cross-stream and time dimensions are processed. The first approach concentrates on cross-decoder dimension and then considers the time dimension, but phone sequence modeling is somehow lost in the way. The second approach runs the opposite route: it can be seen as a phonotactic system (whose factory equipment includes phone sequence modeling) enhanced with additional n-gram co-occurrence modeling. This explains why the second approach provided the best performance among single systems (specially in terms of C_{LLR}), its DET curve being close to that of the optimal fusion (see Figure 6). However, the baseline system provides phone sequence information not present in the first approach, so they complement each other well, and explains why the second approach did not significantly outperform the first approach when fused with the baseline system. Finally, adding systems based on 3-decoder co-occurrences did not improve performance (two last rows in Table 2); instead, it led to worse performance, which may be due either to a mismatch between development and evaluation datasets, or more probably to overfitting in the estimation of fusion parameters. For an easier comparison of system performances, EER and C_{LLR} graphs are shown in Figure 7.

6. Conclusions and Future Work

Two approaches aiming to use cross-decoder phone co-occurrence information (for combinations of $k = 2$ and $k = 3$ decoders) in SVM-based phonotactic language recognition have been proposed and evaluated. The proposed approaches rely on the assumption that cross-decoder co-occurrence information is somehow specific to each target language. The proposed approaches do not involve hard computations; they represent just a means to extract more information from existing decodings.

Systems based on 2-decoder co-occurrences outperformed the baseline system in language recognition experiments on the LRE2007 database. The system based on counts of 2-decoder co-occurrences of phone n-grams yielded the best performance among all single systems, with 2,24% EER (16,73% relative improvement with regard to baseline 2,69% EER), and C_{LLR} = 0,3223 (19,04% relative improvement with regard to baseline C_{LLR} = 0,3981). However, when using 3-decoder configurations, both approaches showed a poor performance compared to the baseline system. This may reveal robustness issues related to: (1) significant differences in phone border detection (Approach 1) which make transitional segments to be dominant, thus producing noisy

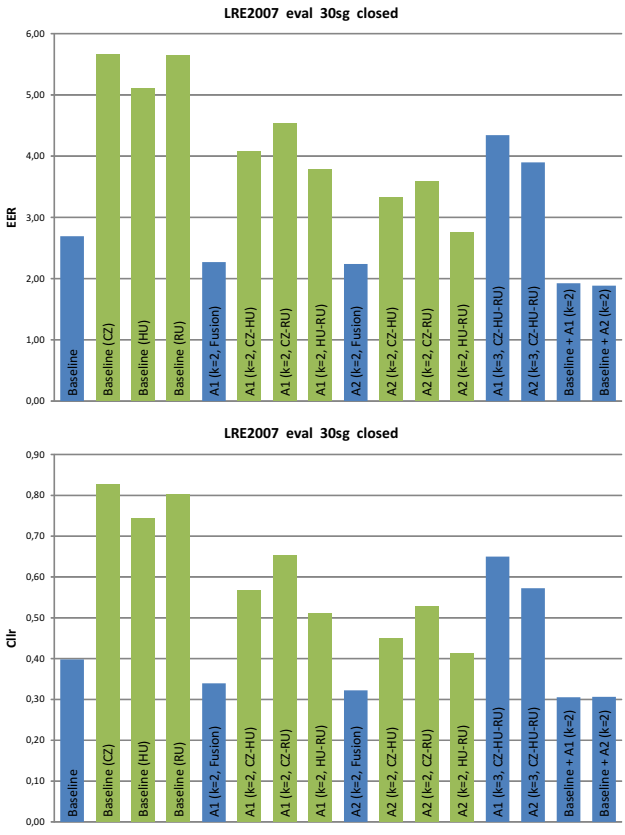

Figure 7: EER and C_{LLR} graphs of some of the language recognition systems evaluated in this work.

sequences of phone co-occurrences; and (2) a huge number of phone n-gram combinations (Approach 2), whose statistics cannot be robustly estimated or that cannot be suitably covered with the 100000 highest counts.

When considering fusions, combining the baseline system with a system based on 2-decoder co-occurrences provided best results, with no significant differences between approaches 1 and 2. The best fused system (Baseline + Approach 2 ($k = 2$)) yielded 1,88% EER and C_{LLR} = 0,3064 (meaning 30% and 23% relative improvements, respectively).

We are currently working on various co-occurrence selection schemes, with the aim to reduce the size of SVM vectors while keeping or even improving performance. Future work will focus on increasing the robustness of phonotactic approaches that integrate time and cross-stream dependencies, specially when using $k \geq 3$ decoders.

7. References

[1] Mikel Penagarikano, Amparo Varona, Luis Javier Rodriguez-Fuentes, and German Bordel, "Using cross-decoder phone coocurrences in phonotactic language recognition," in *Proceedings of the 35th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2010)*, Dallas, Texas (USA), 2010.

[2] M.A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, pp. 31–44, January 1996.

[3] 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.

[4] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 42–54, January 2000.

[5] N. Brümmer and D.A. van Leeuwen, "On calibration of language recognition scores," in *Proceedings of Odyssey - The Speaker and Language Recognition Workshop*, 2006, pp. 1–8.

[6] N. Brümmer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. van Leeuwen, P. Matejka, P. Schwarz, and A. Strasheim, "Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2072–2084, 2007.

[7] Q. Jin, J. Navratil, D.A. Reynolds, J.P. Campbell, W.D. Andrews, and J.S. Abramson, "Combining cross-stream and time dimensions in phonetic speaker recognition," in *ICASSP*, 2003, vol. IV, pp. 800–803.

[8] Christopher White, Izhak Shafran, and Jean-Luc Gauvain, "Discriminative classifiers for language recognition," in *Proceedings of ICASSP*, 2006, pp. 213–216.

[9] Petr Schwarz, *Phoneme recognition based on long temporal context*, Ph.D. thesis, Faculty of Information Technology BUT, Brno, CZ, 2008.

[10] P. Matejka, L. Burget, O. Glembek, P. Schwarz, V. Hubeika, M. Fapso, T. Mikolov, and O. Plchot, "BUT system description for NIST LRE 2007," in *Proc. 2007 NIST Language Recognition Evaluation Workshop*, Orlando, US, 2007, pp. 1–5, National Institute of Standards and Technology.

[11] P.A. Torres-Carrasquillo, E. Singer, W.M. Campbell, T. Gleason, A. McCree, D.A. Reynolds, F. Richardson, W. Shen, and D.E. Sturim, "The MITLL NIST LRE 2007 language recognition system," in *Interspeech*, 2008, pp. 719–722.

[12] F. Richardson and W. Campbell, "Language recognition with discriminative keyword selection," in *ICASSP*, 2008, pp. 4145–4148.

[13] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008, Software available at http://www.csie.ntu.edu.tw/ cjlin/liblinear.

[14] C.C. Chang and C.J. Lin, *LIBSVM: a library for support vector machines*, 2001, http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[15] Niko Brümmer and Johan A. du Preez, "Application-independent evaluation of speaker detection," *Computer Speech & Language*, vol. 20, no. 2-3, pp. 230–275, 2006.