Odyssey 2010
The Speaker and Language Recognition Workshop
28 June – 1 July 2010, Brno, Czech Republic

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.

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.

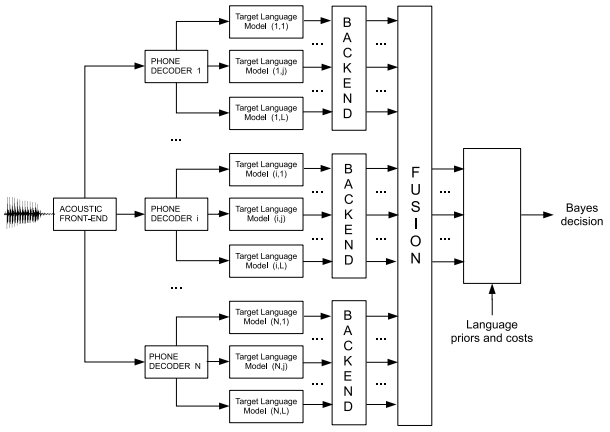

Figure 1: A phonotactic language recognition system.

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).

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.

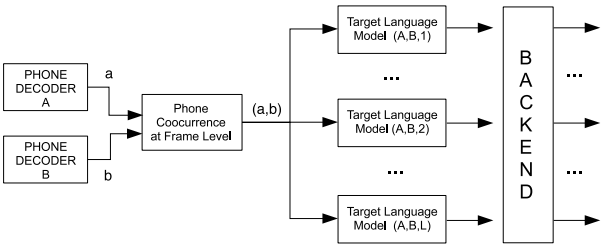

Figure 2: A 2-decoder phone co-occurrence language recognition subsystem.

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 $w_n^{(d)}(t, i)$

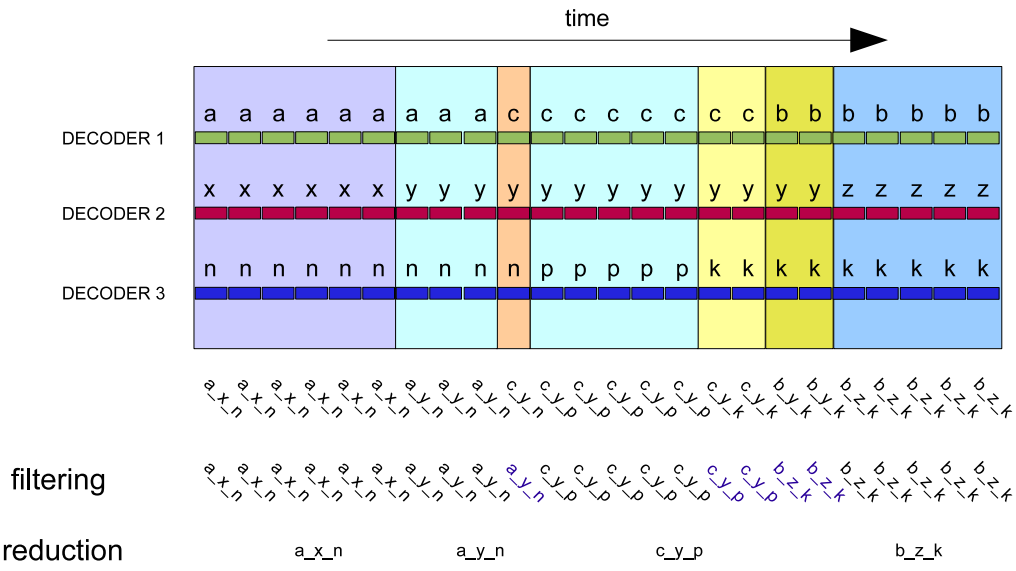

Figure 3: Approach 1 (3-decoder configuration): (1) phone co-occurrence labels are built by concatenating phone labels on a frame-by-frame basis; (2) to handle transitional segments, a mode filter is iteratively applied (until convergence) on a sliding window of 7 frames centered on the analyzed frame;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

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