Anchor-model fusion for language recognition

Ignacio Lopez-Moreno, Daniel Ramos, Joaquin Gonzalez-Rodriguez and Doroteo T. Toledano

ATVS Biometric Recognition Group, C./ Francisco Tomas y Valiente 11,
Universidad Autonoma de Madrid E-28049 Madrid, Spain
ignacio.lopez@uam.es

Abstract

State-of-the-art language recognition systems usually combine multiple acoustic and phonotactic subsystems. The outputs of those systems are usually fused in different ways but the score from a trial is always obtained from N scores from N subsystems. In this paper, a robust novel approach to subsystem fusion in language recognition is proposed based on the relative performance of each trial not just to the claimed model but to all available models. The proposed technique exploits the relative behavior of a given speech utterance over the cohort of anchor models from the different subsystems, resulting in the proposed anchor-model fusion. Experiments fusing seven phone-SVM subsystems submitted by the authors to NIST LRE 2007 assess the robustness to non-uniform data availability over rule-based and trained fusion schemes as linear kernel SVM, as well as significant improvements in performance both in average EER and Cavg as used in NIST LRE.

1. Introduction

Language recognition research has increased in the last few years motivated by demands such as security or human-machine communication. In this multi-class identification problem, a given test speech utterance spoken in an unknown language is classified into one of n classes corresponding to n languages. This task can be either close set (i.e all the languages are known) or open set (i.e the utterance can be in an unknown language). This multi-target problem can be transformed into n *detection* or *verification* problems [1], where the test utterance is compared to a given language model i and a score is generated. This score will be greater to support that the language spoken in the utterance is i. The subdivision of n-class language identification into n language detection problems has been the main task in Language Recognition Evaluations (LRE) conducted by the National Institute of Standards and Technology (NIST) [2].

Nowadays, state-of-the-art language recognition systems are usually combinations of many individual subsystems. This combination allows the final system to efficiently use the complementary information of every subsystem involved in order to improve the individual performance. Such combinations are known as fusion, and they can be divided into ruled based and machine learning based. For a given detection task, in rule-based fusion schemes the final detection score is a combination of the scores coming from the individual subsystems using operations such as product, sum, maximum, minimum, etc. On the other hand, in machine learning fusion, the scores from the individual systems are seen as features to a new classifier, which yields the final score. Machine learning schemes tend to be supervised, needing training data to compute the fusion parameters for the fusion rule or the machine learning. This training scheme may be unique for all trials in the system or dependent on which target language is considered for detection.

This work was funded by the Spanish Ministry of Science and Technology under project TEC2006-13170-C02-01.

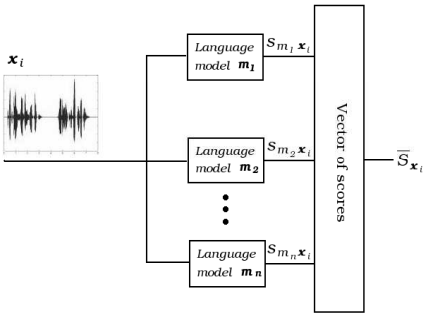


Figure 1: *Schematic of n-class parallel language detection problem. $\bar{S}_{\boldsymbol{x}_i}$ stacks the similarities of $\boldsymbol{x}_i$ over the set of m_j.*

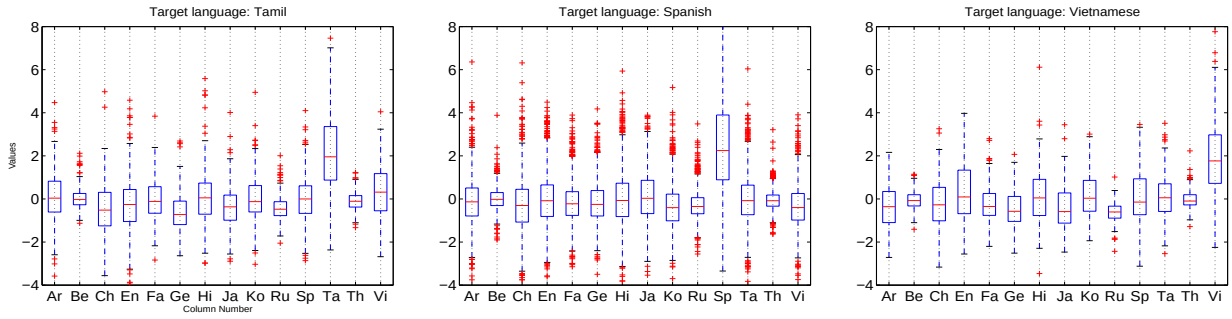
The anchor model space mapping has been successfully used in speaker recognition for speaker recognition and speaker indexing in large databases [3] [4]. This paper, however uses anchor model space for fusion, which represent a novel, machine learning based and target-dependent, fusion scheme for language detection. We can call it anchor model fusion (AMF). A similar idea has already been applied in [5], but using a Gaussian back-end. AMF makes use of a set of *pre-trained* language models (the anchor models), which consists of all the target language models of all the subsystems to fuse. The proposed technique exploits the relative behavior of a given speech utterance over the cohort of anchor models from the different subsystems. The relative behavior of one language versus the others is then modeled by using an *SVM* model [6] for each one of the languages involved in the task.

In order to show the adequacy of the technique, we present results comparing anchor-model fusion to other fusion schemes such as sum rule or linear SVM classifiers. Reported results are obtained over the NIST LRE 2007 protocol. They show a 33.24% and 48.51% relative improvement on C_{avg}, obtained over the sum and SVM fusion schemes respectively.

This work is organized as follows. Section 2 analyzes the motivation of this work, section 3 shows the adequacy of anchor models for language recognition. AMF is presented in section 4. Linear kernel SVM and sum fusion techniques are detailled in section 5. Finally, experiments and results are presented in 6.

2. Motivation

In most language recognition systems, a given speech utterance from a given spoken language $i \in \{1, .., n\}$ can be compared



(a) (b) (c)

Figure 2: *Distributions of scores in the form of box plots over* 14 *detection tasks against pretrained models* m_j, $j \in \{$*arabic, bengali, chinese, english, farsi, german, hindustani, japanese, korean, russian, spanish, tamil, thai, vietnamese*$\}$ *for Tamil (a), Spanish (b) and Vietnamese (c) utterances.*

against each one of the n *pre-trained* target language models m_j, $j \in \{1, .., n\}$. Each comparison, or *trial*, generate a score $s_{m_j,\mathbf{x}_i}$. For every single utterance we can obtain a n dimensional vector $\bar{S}_{\mathbf{x}_i}$ which stacks every single $s_{m_j,\mathbf{x}_i}, j \in \{0, .., n-1\}$ (Fig. 1).

A probability density function (pdf) can be assigned to vectors $\bar{S}_{\mathbf{x}}$ from the same language, namely p_i. This pdf is determined not only by the target pre-trained model m_j of language i ($j = i$), but also by the whole set of non-target language models considered by the system m_j, $j \neq i, j \in \{0, .., n-1\}$. If p_i is different from those of other languages, it can be learned in advance, by a model m'_j. This model is different from the pre-trained model m_j and takes as inputs the scores produced by these pre-trained models.

Figure 2 shows the probability density functions of the scores $s_{m_j,\mathbf{x}_i}$, in three different examples of target languages i, over a set of 14 pre-trained language models m_j. As scores coming from each model m_j constitutes a detection task different of the rest, we show 14 unidimensional pdf per target language instead of a single 14-dimensional pdf. This example illustrates the motivation of the proposed AMF approach. For example, pdfs from utterances x_i where i=Vietnamese (figure 2 (c)) show that, as expected, target scores where $i = j$ =Vietnamese tend to be higher than non-target scores where $i \neq j$. Moreover, scores from different pre-trained models m_j present a different behavior. For instance, for j =korean scores tend to be higher than the obtained when j =Russian. This behavior can be learned by some data driven model m'_{vi}, whose input data will be the scores generated by utterances $\mathbf{x}_i$ when i =Vietnamese.

Thus, the AMF approach assumes that the information of the behavior of the scores over all the m_j pre-trained models from different baseline systems, namely $\bar{S}_{\mathbf{x}_i}$, should help to improve performance over the use of a single detection score $s_{m_i,\mathbf{x}_i}$.

An interesting property of this space is that if data for training a language model is scarce, working with the whole set of scores should improve robustness, as new infomation is incorporated about the relative behavior of language model m_j against the rest of languages in the system.

3. Anchor models for language recognition

Anchor model space projection is a function that maps each speech utterance from the original, or *source*, feature space into a new *anchor model* space. The dimensions of this new space are similarity scores of each speech utterance over a previously trained model in the cohort of anchor models. The whole cohort will be denoted as $\mathbf{m} = \{m_1 \cdots m_N\}$. This similarity space allows learning the behavior of the speech utterance $\mathbf{x}$ with respect to $\mathbf{m}$ by obtaining its similarity scores vector $\bar{S}_x$.

The N pre-trained models in $\mathbf{m}$ can be generated using techniques such as Gaussian Mixture Models (GMM), SVM, n-grams, etc [7]. Thus, in an anchor model approach, from each $\mathbf{x}$ and for a given cohort $\mathbf{m}$, we can obtain a vector of similarity scores

$$\bar{S}_{\mathbf{x},\mathbf{m}} = [s_{\mathbf{x},m_1} \cdots s_{\mathbf{x},m_N}] \quad (1)$$

that stack the individual similarities of $\mathbf{x}$ over each one of the models m_j in $\mathbf{m}$ (Fig. 1). The anchor model space represents vectors $\bar{S}_{\mathbf{x},\mathbf{m}}$. Thus, a new model m'_j of the speech pattern can be generated in the anchor model space from $\bar{S}_{\mathbf{x},\mathbf{m}}$ using any data driven technique such as GMM, SVM, etc.

Therefore, the final anchor model space is defined by a cohort of anchor models $\mathbf{m}$ and its similarity functions $s_{m_j}(\cdot)$ used by each m_j.

The size N of $\mathbf{m}$ defines the anchor model dimensionality. If N increases, the probability of finding a characteristic behavior of the speech pattern in the anchor model space increases too. In real speaker and language recognition systems, N is limited by the amount of available languages and the computational complexity.

Similarity functions $s_{m_j}(\cdot)$ defines the distance criteria in the anchor model space. It is common that the technique used to build the models in $\mathbf{m}$ determines the similarity function to use. For instance, GMM models usually use statistical similarity criteria while SVM uses algebraic distances based on projections. By using different similarity functions $s_{m_j}(\cdot)$ in $\mathbf{m}$, different and possibly complementary information is introduced in the anchor model space.

4. Anchor Model Fusion (AMF)

In the proposed supervised machine learning AMF scheme, the cohort $\mathbf{m}$ includes the n pre-trained language models of the n-classes problem for each language recognition system to fuse.

$$\bar{S}_{\mathbf{x},\mathbf{m}} = \left[\bar{S}^1_{\mathbf{x},\mathbf{m}}, \cdots \bar{S}^{N_{sys}}_{\mathbf{x},\mathbf{m}}\right] \quad (2)$$

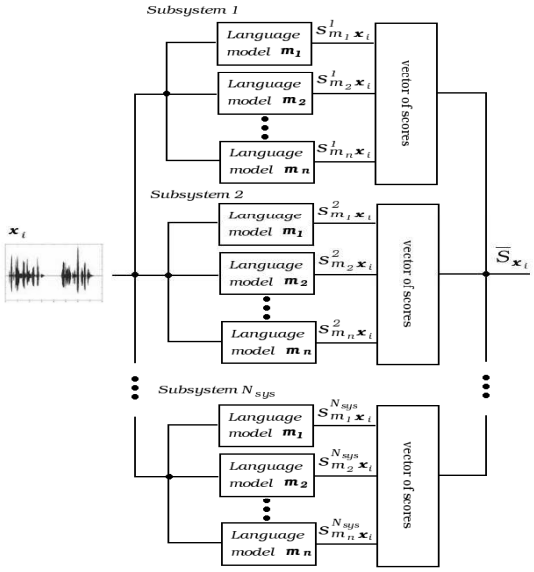


Figure 3: *Schematic diagram of generation of features (scores) in the anchor model space.* $\bar{S}_{\mathbf{x},\mathbf{m}}$ *stacks the similarities of* $\boldsymbol{x}_i$ *over the set of models* m_j^l*, for language* j *and subsystem* l

.

The schematic diagram can be seen in Fig. 3. Hence, anchor model space dimension is $N = n \times N_{sys}$. Where N_{sys} is the number of subsystems.

If for the language recognition subsystem $l \in \{1, .., N_{sys}\}$, we suppose that the model m_j^l in its own source space can recognize the language j with some performance rates. New models m'_j, in the anchor model space can ideally improve the performance rates of the baseline systems due to the final similarity score is a combination of all similarities, including $s^l_{m_j,\mathbf{x}_i}$, $s^l_{m_{j'},\mathbf{x}_i}$ ($j' \neq j$) and its relation singularities. Performance is only improved when this relation singularities differs from a language, and from a system, to another.

Note that this scheme requires two steps for each language model. The first one trains the models m_j^l in **m**, and the second one trains the final language model $m_j^{'}$ in the anchor model space.

5. Other fusion schemes for language detection

5.1. SVM fusion

Linear-kernel SVM fusion is a supervised machine learning based scheme, which works as follows. For a given trial, or combination (m_j vs $\mathbf{x}_i$) , we can obtain a vector that stacks the N_{sys} scores of all the subsystems to fuse.

$$\bar{S}_{\mathbf{x},m_j} = \left[s_{\mathbf{x},m_j,1} \cdots s_{\mathbf{x},m_j,l_{N_{sys}}} \right] \tag{3}$$

where $s_{\mathbf{x},m_j,l}$ is the similarity score of the language model m_j, the subsystem l over the text utterance **x**. Thus, we can obtain a language-dependent fusion SVM model $(\bar{\mathbf{w}}_i, b)$ by labelling target score vectors as belonging to class $\omega = 1$ and to class $\omega = -1$ otherwise. Unknown vectors $\bar{S'}_{\mathbf{x},m_j}$ are scored as:

$$f(\bar{S}_{\mathbf{x},m_j}) = \bar{S}_{\mathbf{x},m_j} * \bar{\mathbf{w}}_i + b \tag{4}$$

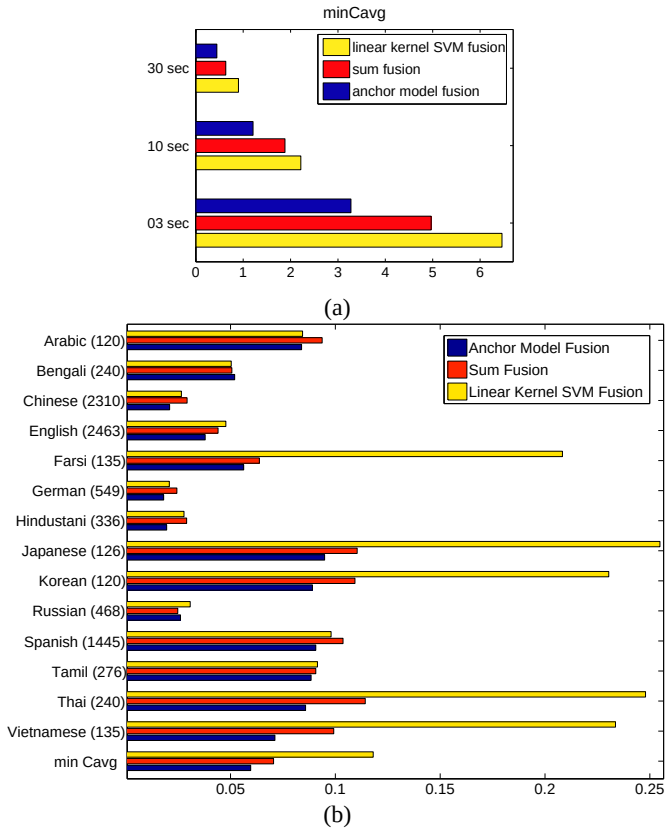


Figure 4: C_{avg} *for anchor model, sum and linear kernel SVM fusions. Results are divided by test segment durations (a), and detailled per language for 30 sec. test utterances (b). Brackets show the number of utterances used to train each model.*

Notice that in linear kernel SVM fusion scheme, the similarity vectors depends on the given trial, and its dimensionality is N_{sys}. It differs to AMF approach where similarity vectors depend on the testing utterance. Its dimensionality is $N_{sys} * n$.

5.2. Sum fusion

AMF scheme is also compared with the popular rule based sum fusion. For each trial, the final score is the sum of all the subsystem scores:

$$s_{\mathbf{x},i} = s_{\mathbf{x},m_j,1} + \cdots + s_{\mathbf{x},m_j,N_{sys}} \tag{5}$$

The main advantage of sum fusion relies in its simplicity, since no training phase is required. However, its performance is seriously degraded if the score range of the subsystems is heterogeneous. Therefore some kind of score pre-normalization is required.

6. Experiments

Experiments have been performed using the evaluation protocol proposed by NIST in its 2007 Language Recognition Evaluation (LRE). The database used in this evaluation consists of a significant subcorpus provied by NIST for testing purposes. Language recognition is evaluated as 14 different detection sub-problems, one per language. This multi-detection task is referred to as the *general* condition. Test utterances are also divided in three

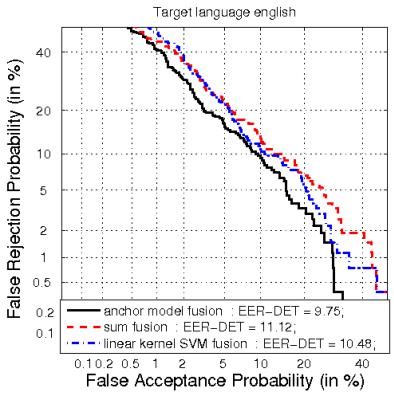


Figure 5: *Results for the three fusion schemes tested: linear kernel SVM, sum fusion and AMF. DETs are shown for english model ($m'_{english}$) and 30 sec. test segments.*

different durations, namely 30, 10, and 3 seconds of speech on average after silence removal. Details can be found in [2].

AMF was used for fusing 7 phonetic subsystems submitted by ATVS-UAM to NIST LRE 2007. Each subsystem works as follows. First, speech utterances are divided into speech and silence segments. Speech segments are recognized with an open-loop phonetic decoder. Phonemes are extracted through the most probable transcription. Probabilities from frequency counts of unigrams, bigrams, and trigrams were used as input vectors for training a linear kernel SVM classifier per target language. These subsystems are 7 phonetic decoders from the following languages: English, German, French, Arabic, Basque and Russian. Details about this phone-SVM technique can be found in [8]. Prior to apply fusion, similarity scores of every single subsystem were T-normalized.

Each one of the 14 language models of every phonetic subsystem are used to built the anchor model cohort **m**. Therefore, the anchor model space dimension is $14 \times 7 = 98$. After mapping each speech utterance in the anchor model space, a linear kernel SVM classifier is used for training each language model from vectors $\bar{S}_{\mathbf{x},\mathbf{m}}$ of scores. The background set for system tuning is a subset of databases from previous NIST LRE from years 1996, 2003 and 2005; as well as the Callfriend database[1]. Linear kernel SVM fusion and sum fusion of the 7 phonetic systems were also tested in the same development data, and will serve as baseline results for the AMF system.

	Test segment duration		
EER_{avg}	$30sec$	$10sec$	$03sec$
Anchor Model Fusion	**7.36**	**13.82**	**23.52**
Sum Fusion	8.28	14.90	23.61
Linear SVM Fusion	12.66	19.40	27.73

Table 1: EER averaged over all the languages involved in LRE 07, detailed per fusion scheme and test segment duration

6.1. Results

This section present the performance measurements of the systems tested. Results are shown as C_{avg}, global C_{avg}, averaged EER, and language dependent DETs curves. A sample DET for the english model and 30 seconds test utterances is shown in Fig. 5.

Figure 4 (a) and table 1 shows the global performance of the three systems tested. In all cases the anchor model fusion outperforms the other two systems evaluated, sum and linear kernel SVM fusion. In EER_{avg} the relative improvement of the AMF performance increases with the duration of the test segment. From 0.38%, for 3 seconds segments, to 11.12% for 30 seconds related to the sum fusion. For the linear kernel SVM the relative improvement goes from 15.18% (3 sec.) to 41.86% (30 sec.). However, in C_{avg} the relative improvement of the AMF is more notorious and constant with the duration of the test segment. $\approx 30\%$ related to the sum fusion, and $\approx 50\%$ related to the linear kernel SVM for all durations.

Figure 4 (b) details the results per language in 30 seconds test segments. It shows that linear kernel SVM fusion is a non efficient solution for some languages with scarce training data. It also shows that AMF does not suffer this problem maybe because AMF models seize more information from each utterance.

7. Conclusions

This paper presents a novel fusion technique for a n-class recognition problem such as language identification that we call Anchor Model Fusion. This data driven technique is based on the anchor model approach and allows to seize the relations in the similarities among all the language models of all the systems. AMF results outperform the other fusion techniques analyzed (sum and linear kernel SVM fusion). Future work will analyze an efficient pruning of the cohort of anchor models and the generation of vectors $\bar{S}_{\mathbf{x},\mathbf{m}}$ at the frame level.

8. References

[1] N. Brümmer and D. van Leeuwen, "On calibration of language recognition scores," in *Proc. of Odyssey*, San Juan, Puerto Rico, 2006.

[2] Alvin F. Martin and Audrey N. Le, "Nist 2007 language recognition evaluation," in *Proc. of Odyssey*, Stellenbosch, South Africa, 2008.

[3] D.E. Sturim, D.A. Reynolds, E. Singer, and J.P. Campbell, "Speaker indexing in large audio databases using anchor models," Salt Lake City, USA, 2001, vol. 1, pp. 429–432.

[4] M. Collet, Y. Mami, D. Charlet, and F. Bimbot, "Probabilistic anchor models approach for speaker verification," 2005, pp. 2005–2008.

[5] Bin Ma, Haizhou Li, and Rong Tong, "Spoken language recognition using ensemble classifiers," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2053–2062, Sept. 2007.

[6] B. Schölkopf, C. J. C. Burges, and A. J. Smola, *Advances in Kernel Methods and Support Vector Learning*, MIT Press, 2000.

[7] Jacob Benesty, M. M. Sondhi, and Yiteng (Eds.) Huang, *Springer Handbook of Speech Processing. Part G*, Springer, 2008.

[8] D.T. Toledano, J. Gonzalez-Domingez, A. Abejon, and D. Spada, "Improved language recognition using better phonetic decoders and fusion with mfcc and sdc features," in *Proc. of Interspeech*, Antwerp, Belgium, 2007, pp. 194–197.

[1] Details can be found in the LDC website: www.ldc.upenn.edu.