Integration of acoustic information and PPRLM scores in a multiple-Gaussian classifier for Language Identification

R. Córdoba, R. San-Segundo, J. Macías, Juan M. Montero, R. Barra, L.F. D'Haro, J.C. Plaza, J. Ferreiros

Speech Technology Group. Dept. of Electronic Engineering. Universidad Politécnica de Madrid
E.T.S.I. Telecomunicación. Ciudad Universitaria s/n, 28040-Madrid, Spain

Abstract

In this paper, we present several innovative techniques that can be applied in a PPRLM system for language identification (LID). We will show how we obtained a 53.5% relative error reduction from our base system using several techniques. First, the application of a variable threshold in score computation, dependent on the average scores in the language model, provided a 35% error reduction. A random selection of sentences for the different sets and the use of silence models also improved the system. Then, to improve the classifier, we compared the bias removal technique (up to 19% error reduction) and a Gaussian classifier (up to 37% error reduction). Finally, we included the acoustic score in the Gaussian classifier (2% error reduction) and increased the number of Gaussians to have a multiple-Gaussian classifier (14% error reduction). We will show how all these improvements are remarkable as they have been mostly additive.

1. Introduction

Automatic language identification (LID) has become an important issue in recent years in speech recognition systems. Multilinguality is a must for many systems, so the language of the caller has to be identified as soon as possible in order to use the appropriate recognition system specific to that language.

To do language identification, first we have to identify which factors are more critical to distinguish between languages. We can identify several factors of differentiation: the realization of allophones and sounds (some allophones exist in one language but not in other languages) and information related to the sequence of allophones, which has demonstrated to be vital: some sequences of allophones do not exist in one language (or occur very little), so the identification of those sequences is crucial for LID. Another possibility is to use prosodic features – fundamental frequency, duration and/or energy – as the intonation may differ drastically between languages.

Many techniques have been suggested in recent years for this task. The most widespread technique is the phone-based approach, like Parallel phone recognition followed by language modeling (PPRLM) [1][2], which classifies languages based on the statistical characteristics of the allophone sequences and has a very good performance.

Another popular technique is a simple GMM classifier. This technique addresses the first differential factor between languages: every language has sounds that are specific to it. Its main advantage is that we do not need labeled data to train the classifier, so it is a very cheap system. Its main drawback is its low performance, due to the fact that it does not deal with any information regarding the sequence of sounds (the second main factor of differentiation between languages.) In recent years, some techniques have been proposed that try to take the advantages from both techniques: a GMM classifier called "GMM tokenizer" [3][4][5]. In this approach, the output of the classifier (for each frame, the tokenizer outputs the index of the Gaussian component scoring highest in the GMM computation), is used as input to a "language model" (LM) module, where the sequence of the different indexes is learnt. This technique uses both acoustic information and sequence information, so it seems to be suitable and has the same advantages as the GMM alone: labeled data is unneeded and it is faster that the phone-based approaches. Nevertheless, in all previous studies the performance of this technique is worse than PPRLM, but has one advantage: the combination of PPRLM and this technique improves the overall result. So, it offers complementary information to the task, but with the cost of CPU time due to the use of PPRLM.

Another possibility is to base the identification on the score given by a full continuous speech recognizer. As we demonstrated in [6], the results obtained with this technique are probably the best that can be obtained, as it models both acoustic and phonetic information, together with the sequence of allophones and words, but it has some important disadvantages: a complete speech recognition system has to be trained, a lot of labeled data is needed and it would be difficult to have a real-time system for several languages. In any case, for the identification of two languages, which can be enough for many applications and/or countries, it is the best option. In [7] a full recognizer is also proposed and the recognizer scores are normalized and compared with a linear classifier.

An interesting variant of PPRLM is presented in [8] with several proposals: different ways to combine the

1-4244-0472-X/06/$20.00 © 2006 IEEE

allophone sequence information with the acoustic models, use of durations (prosodic information) and a tree-based language model. It is remarkable the integration of several sources of information.

Another technique is to use a lattice instead of the allophone sequence [9] and a neural network at the output of the classifier, instead of doing the average of the scores. This way, there is an improvement in the classifier. In our paper we propose a Gaussian classifier instead of the neural network.

We should also mention the proposal in [10]: use PPR, include bias removal to improve the classification, and include acoustic and allophone sequence information in the classifier, using a Gaussian classifier similar to the one proposed in this paper.

In summary, there is a general agreement that PPRLM is the best option if you look for performance and have labeled data available to model the phone recognizers. In fact, it has been widely used for speaker recognition with very good results [11], especially in mismatch conditions.

This paper is a continuation of the work done in [2]. We are going to focus now on improving the classifier, using bias removal and a multiple-Gaussian classifier mixing acoustic and allophone sequence information.

This work has been done under project INVOCA, for the public company AENA, which manages Spanish airports and air navigation systems [12].

The paper is organized as follows. We present the database used and the experimental setup in Section 2. A brief overview of the PPRLM technique is given in Section 3. Then, in Section 4 we present three initial approaches to improve the base system. In Section 5, we focus on the classifier, comparing the bias removal technique with the Gaussian classifier. In Section 6, we include acoustic information and increase the number of Gaussians in the Gaussian classifier. The conclusions are given in Section 7.

2. System setup

2.1. Database

We use a continuous speech database, which consists of very spontaneous conversations between controllers and pilots. For speech recognition it is a very difficult task, noisy and very spontaneous, as in “lufthansa four two seven nine start up approved clear to frankfurt standard departure somosierra one echo three six left squawk one zero two three report parking position”.

We have one big drawback with the database: all speakers are native Spanish. So, many of them do not reflect all the phonetic variations in English. This is a decisive factor in all cases for English identification. We have a second drawback: the controllers use to mix Spanish for greetings and goodbyes even when the rest of the sentence is in English. Also, many company names and airports have the Spanish pronunciation embedded in the English conversation.

In Table 1 we can see the contents of the database in sentences and hours of speech. We have experimented with different divisions of the training set sentences to train the acoustic HMMs and to train the language models. So, we will not include it here, but some comments are included regarding this division in the respective Sections.

Table 1. Database (sentences / hours)

	Spanish	English
Training set	5,529 / 8.0	3,153 / 5,7
Validation set	500 / 0.9	453 / 0.9

In the test set, we have considered sentences with a minimum of 0.5 sec., and a maximum of 10 sec., with an average duration of just 4.5 sec. This is another limitation in our system: we have to identify the language using less than 5 seconds of speech.

2.2. General conditions of the experiments

The system uses a front-end with PLP coefficients derived from a mel-scale filter bank (MF-PLP), with 13 coefficients including c0 and their first and second-order differentials, giving a total of 39 parameters per frame.

For the phone recognizers, we have used context-independent continuous HMM models. For Spanish, we have considered 49 different allophones and, for English, 61 different allophones. So, we have tried to cover all possible phonetic variations in both languages, specially including allophones that do not exist in the other language. All models use 10 Gaussians densities per state per stream.

3. Description of PPRLM

The main objective of PPRLM (Parallel Phone Recognition Language Modeling) is to model the frequency of occurrence of different allophone sequences in each language. This system has two stages. In the first stage, a phone recognizer takes the speech utterance and outputs the sequence of allophones corresponding to it. The sequence of allophones generated by the phone recognizers is used as input to a language model module. In the second stage, the language model module scores the probability that the sequence of allophones corresponds to the language.

It can use several phone recognizers modeled for different languages. The advantage is that using many recognizers we can cover most of the phonetic realizations of the languages. Its main drawback is speed: processing time is multiplied by the number of recognizers. Using PPRLM, we can even have phone recognizers modeled for languages different than the languages that have to be identified, but obviously if there is a match between the input language and the language of the models the performance will be better, because you can model explicitly the phonetic variations of each language. In our case, as we want to identify English and Spanish and we have labeled data for both of them, the best option is to use PPRLM with phone recognizers trained for English and Spanish.

In the identification stage a language model module scores the probability that the sequence of allophones corresponds to the language according to the process illustrated in Figure 1. The overall score is calculated as an average between both scores obtained for the same language according to (1). Interpolated n-gram language models are used to approximate the n-gram distribution as the weighted sum of the probabilities of the n-grams considered. In our case, we have considered up to trigrams. For a sequence of three consecutive symbols observed in the phone stream, we use the formula (2).

$$SC-CAST=\frac{SC0+SC2}{2}\ ;\ SC-ING=\frac{SC1+SC3}{2} \qquad (1)$$

$$\begin{aligned} S\left(w_t, w_{t-1}, w_{t-2}\right) = \alpha_3 \cdot P\left(w_t \mid w_{t-1}, w_{t-2}\right) + \\ \alpha_2 \cdot P\left(w_{t-1} \mid w_{t-2}\right) + \alpha_1 \cdot P\left(w_{t-2}\right) + \alpha_0 \cdot P_0 \end{aligned} \qquad (2)$$

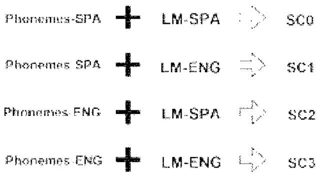

Figure 1. PPRLM Score average

3.1. Results presentation

In all our experiments we have obtained the results for all possible combinations of weights α_1, α_2, and α_3, in 0.1 steps. Throughout the paper we will present the results for the average of all weight combinations (Average column in the tables) and for the best result (Minimum column), because in some cases, especially with the best systems, the improvements may be low for the best combination of weights, but the technique may be very promising as it works better in average for all weight combinations. In general, best (minimum) results occur with the biggest contribution from the trigram score, reflecting that the trigram is the most discriminative feature for language identification (if it is estimated correctly, of course). In all tables, we present in parenthesis the relative improvement in relation to the base system considered.

4. Initial improvements to the base system

4.1. New distribution of the database

One important conclusion in [2] was that the database had a bad distribution, as it dedicated a very small set to the training of the language models. So, we decided to dedicate 50% of the training material to train the acoustic HMMs and 50% to train the language models.

The average improvement with this approach is **13.5%**, showing that, as we assumed, the amount of data dedicated to train the HMMs is not critic.

4.2. Threshold in score computation

As the size of the database is small, there is quite a big number of trigrams that do not have enough training samples and, so, their estimates are not reliable. We tested several alternatives for language model smoothing (Katz smoothing and Backoff Kneser-Ney smoothing), but the results were very similar, showing little improvement.

We decided to apply a fixed threshold or additive factor to the score value, in a similar way to the variance flooring applied in HMM estimation: use as the minimum variance a fraction of the average variance in the whole database.

The objective of this additive factor is to give more importance to the allophone sequences that have a high probability in one language and, at the same time, reduce the effect of sequences that have not appeared in training. This way, we give more relevance to sequences that are really specific of one language, and do not 'spoil' the score with intermediate values from less relevant sequences.

We considered three alternatives, in all cases working in the log domain:

1) **Fixed and common additive factor**. We propose the following formula for the score (the logarithmic implementation of equation (2)):

$$S(F)=10\log\left(\prod_{i=0}^{N} P_i(F)\right)=-\sum_{i=0}^{N} 10\cdot\alpha_i\cdot\log\left(P_i(F)+\beta\right) \qquad (3)$$

where N is the order of the N-gram, α_i is the weight for the ith n-gram and $P_i(F)$ is its probability. β is the additive factor. Several values were tested, being the optimum 0.01.

2) **n-gram specific fixed additive factor**. The β is now n-gram specific:

$$S(F)=10\log\left(\prod_{i=0}^{N} P_i(F)\right)=-\sum_{i=0}^{N} 10\cdot\alpha_i\cdot\log\left(P_i(F)+\beta_i\right) \qquad (4)$$

The optimum values were β_{uni}=0.027, β_{bi}=0.04 y β_{tri}=0.08. Obviously, we do not like this approach because the β value is too empiric.

3) **Variable additive factor**. We decided to apply an additive factor which was dependent on the average scores in the language model. In the following formula, $\overline{p}_i$ is the average of all probabilities for the ith n-gram, and λ is a smoothing factor.

$$S(F) = -\sum_{i=0}^{N} 10\cdot\alpha_i\cdot\log\left(P_i(F) + \frac{\overline{p}_i}{\lambda}\right) \qquad (5)$$

Several experiments were run to estimate the optimum λ factor. We are glad to say that very little differences in performance were observed using λ values between 4 and 8.

In Table 2 we can see the results obtained with the 3 additive factors. In parenthesis we can see the relative improvement in relation to the 'None' system. As we describe in Section 3.1, we present the results for the average of all weight combinations (Average column) and for the best result (Minimum column).

Table 2. Results for different additive factors

Threshold technique	Average	Minimum
None	8.27	6.80
Fixed	7.95 (4.0%)	6.27 (7.8%)
n-gram specific	7.70 (6.9%)	5.84 (14.1%)
Variable	6.26 (24.3%)	4.46 (34.3%)

As we can see, the improvement is outstanding, showing the suitability of this approach, especially for the third approach. Even though it is simple, it has been the best improvement in this series of experiments.

4.3. Random selection of sentences

Our database consists of conversations between controllers and pilots. So, the same controller uttered a large group of sentences which were sequential in the database until there was a shift change. We were afraid that our system was making some kind of speaker modeling instead of language modeling, as we desired, i.e., our models could be capturing the specific characteristics of the predominant controller instead of the language used. So, we decided to create new lists using a random selection procedure, namely Fisher-Yates. We can see in Table 3 that there is an important improvement of 16.6% in average, showing that in fact there was some sort of implicit speaker modeling.

Table 3. Results with random selection

	Average	Minimum
Original lists	6.26	4.46
Randomly selected	5.24 (16.6%)	4.24 (5.0%)

4.4. Silence models

In our original system, we decided not to consider silence models in the output of the phone recognizer, because they could indicate just noise and do not have a linguistic meaning. Nevertheless, we considered that without the silence we were not estimating important trigrams which are especially relevant for language identification, e.g. 'ai-t-sil' in the word 'flight', which is extremely rare in Spanish. So, we run an experiment considering the silence models with the results shown in Table 4. We can see again that our intuitions were correct, with a remarkable improvement considering that there are very little changes with this approach. The minimum result is obtained using a bigger weight for the trigram, which supports our conclusions. We have to mention that the improvement was even 14% using the original lists instead of the random ones, probably because our system is approaching a top performance.

Table 4. Results with silence models

	Average	Minimum
Randomly selected	5.24	4.24
+ silence models	5.03 (4.0%)	3.92 (7.55%)

5. Improvements to the classifier

5.1. Bias removal

5.1.1. Description

As is described in [10], the general PPRLM has a flaw: there is the possibility of having a bias in the log-likelihood score which is different for the languages considered. This is especially relevant when the phone recognizers have a different number of units. The language with fewer units will have higher probabilities in the LM score (think of the unigram case), and so the classifier will tend to select that language. We had observed that behavior before: in most experiments the error rate was lower for Spanish because the classifier tended to select Spanish. We first thought (as we concluded in [2] and [6]) that it was due to the speakers of the database being native Spanish, but now we are sure that the real reason was this bias effect, as we have 49 phonetic units for Spanish and 61 for English.

To eliminate this bias, two options are proposed in [10]. We have experimented with the first one, which will be used for comparison purposes with the Gaussian classifier proposed in Section 5.2.

The basic idea of bias removal is to use as LM score the original score minus the average of all LM scores in the training database (a language-dependent bias).

5.1.2. Database for bias estimation

The implementation is quite simple, but an important issue that has to be faced is which part of the training database must be used to compute this average value.

We can divide the training database in 3 different sets: the first one is used to train the acoustic models, the second one to train the language models and the third set to estimate the bias value. This could be the optimal option if the database was large enough, as all estimations are independent. The problem is that, as our database is small, when we reduced the size of the first two sets to dedicate some sentences to estimate the bias value, all results worsened due to insufficient training data. So, we had to discard this option.

Another option is to estimate the bias value in the original training sets. We have two options:

1. Estimate the bias with the language models training list. This is the worst option: as the LMs have been estimated using this list, the bias value estimated is not reliable because it is too optimistic.

2. Estimate the bias with the acoustic models training list. Even though this data does not participate in the LM estimation, this could be a dangerous option, because it could have the same undesirable effects as the previous option. But we observed that the LM score distribution in this set was very similar to the score distribution in the test set. So, we decided to use this option with good results.

5.1.3. Results for bias removal

In Table 5 we present the results obtained using bias removal in a system without the improvement described in Section 4.2. We can see an outstanding improvement, showing that this technique is effective when there is an obvious bias in the log-likelihood score as we had presumed.

Table 5. Results for bias removal

	Average	Minimum
No threshold	8.27	6.80
Bias removal	6.98 (15.6%)	5.5 (18.9%)

But we have to admit that the same technique applied to the best system so far – after the threshold technique from Section 4.2 – showed no relevant improvement, just 0-1% relative. The most probable explanation is that the additive factor compensates the bias effect, and the improvements in this case are not additive.

5.2. Gaussian classifier

5.2.1. Description

Another possibility to tackle the issue of different bias in the LM scores is to use a Gaussian classifier instead of the usual decision formula applied in PPRLM (see equation 1 in Section 3). With all the scores provided by every LM in the PPRLM module we prepare a score vector. With all the sentences in the training database we estimate the Gaussian distribution of their respective score vectors for every language. So, we will have a Gaussian distribution for each language in the system.

Now, the recognized language is not the one with the largest average score. The distance between the input vector of LM scores and the Gaussian distributions for every language is computed, and the distribution which is closer to the input vector is the one selected as identified language.

So, the LID problem can be treated as a conventional N-class classification problem (for N languages) in the score space of dimension D (D scores considered in our system). Each class is represented by a Gaussian density N(μ_l, Σ_l), where μ_l and Σ_l are the mean and covariance of class *l*. They are estimated from the training data of P vectors of class l as:

$$\mu_l = \frac{1}{P}\sum_{p=1}^{P} x_{l,p} \qquad (6)$$

$$\Sigma_l = \frac{1}{P}\sum_{p=1}^{P} (x_{l,p} - \mu_l)(x_{l,p} - \mu_l)^t \qquad (7)$$

A test utterance is classified as language *l** based on its score vector *y*, if:

$$d(y, \mu_{l*}, \Sigma_{l*}) \le d(y, \mu_l, \Sigma_l), l = 1,..., N \qquad (8)$$

where,

$$d(y, \mu_l, \Sigma_l) = (y - \mu_l)^t \Sigma_l^{-1} (y - \mu_l) \qquad (9)$$

is the distance measure. We have considered the weighted Euclidean distance (Σ diagonal) instead of a full covariance matrix as we are aware of the insufficient training data to estimate the full matrix.

The advantage of the Gaussian classifier is that it does not suffer from the bias problem as it does not use an absolute discriminant function.

5.2.2. Database for the Gaussian classifier

For the Gaussian classifier, the same considerations as for bias removal can be made (see Section 5.1.2). In this case, the problem addressed is even more notorious, as we need more data to estimate a reliable Gaussian distribution than we need to estimate just the bias in the score. So, again we decided to use the acoustic models training list to estimate the Gaussian distributions.

5.2.3. *Score vector for the Gaussian classifier*

As we have several scores in the PPRLM system, there are several options for the feature vector of scores:

1. Basic. Use the four scores (M acoustic models x N language models, 2 x 2 in our case) shown in Figure 1. This would be the typical option, probably used by most systems where a Gaussian classifier has been considered. The problem with this approach is that there are big variations in these scores even with sentences from the same language, and the result is that the Gaussian distributions estimated are too wide and are not discriminative enough (there is a big overlap between the distributions for the different languages).

2. Individual scores. To overcome the big variations in score, we first considered the possibility to model the distribution of each n-gram in the score computation for our feature vector: the score for unigram, bigram and trigram from equation (2) in Section 3. So, we had a feature vector of dimension 12 (M acoustic models x N language models x 3 n-gram scores). We considered this approach because we observed that individual n-gram scores were a little more homogeneous than the global PPRLM scores. The drawback is that the increase in dimension causes a worse estimation as we still have the same amount of training data.

3. Differential scores. Instead of using absolute values, we considered differential scores, which for every sentence are computed as the difference between the score obtained by the LM of the same language of the acoustic models considered (Spanish-Spanish or English-English) and the score obtained by the other 'competing' language: SC0 – SC1 and SC3 – SC2 in Figure 1. So, this score can be computed both in training and testing. We also considered the differentiation between individual scores: unigram, bigram, and trigram. In Table 6 we can see the summary of parameters for the score vector.

Table 6. Differential score vector

Phonemes-SPA	SCO-SC1 for unigram
	SCO-SC1 for bigram
	SCO-SC1 for trigram
Phonemes-ENG	SC3-SC2 for unigram
	SC3-SC2 for bigram
	SC3-SC2 for trigram

This is an important innovation in this work. We observed that these differential scores are much more homogeneous, being the result that the estimated distributions exhibit a much smaller overlap with the competing language.

In a multiple language system the proposal for the differential score would be:

$$SC_{\text{current language}} - \text{Average}\,(SC_{\text{other languages}})$$

In Table 7, we can see the results for the 3 techniques in a system without the improvement described in Section 4.2. As we can see, the results for the Basic and Individual options are similar, and in both cases there is a remarkable worsening, which can be due to two facts: the great variations in score that we have already mentioned, and the insufficient size of the database. Nevertheless, the results for the Differential scores are outstanding, more than 30% relative, showing the suitability of our approach.

Table 7. Results for the Gaussian classifier

Score vector	Average	Minimum
No threshold	8.27	6.80
Basic	11.43 (-38.2%)	7.7 (-13.8%)
Individual	10.94 (-32.3%)	7.7 (-13.8%)
Differential	5.82 (29.6%)	4.3 (36.8%)

If we apply the technique with the best system so far (see Table 8, only Differential is presented to summarize, although similar conclusions can be extracted for Basic and Individual), the results show a smaller improvement, but are better than for the bias removal technique. Again, the improvement of the threshold technique is not additive with the Gaussian classifier. The results for the 'Average' column cannot be compared because the inclusion of the n-gram weights in equation (2) is now completely different (we have included them in the distance computation as a factor that multiplies the standard deviation considered in the distance measure).

Table 8. Results for the Gaussian classifier

	Average	Minimum
Base	5.03	3.92
Gaussian classifier	5.63 (not comparable)	3.71 (5.41%)

In any case, getting these results is a fantastic starting point, as it is easy to include acoustic and allophone sequence information using this Gaussian classifier. And, as we will see, some further improvements are still feasible if we increase the number of Gaussians in the classifier.

6. Improved Gaussian classifier

6.1. Inclusion of acoustic information

One drawback in PPRLM modeling is that the basic technique only takes into account information regarding the allophone sequence. As we mentioned in the introduction, another techniques as the "GMM tokenizer" provide a good performance using both acoustic and "sequence of sounds" information. But the acoustic score of the phone recognizers cannot be included in the basic PPRLM formula (equation (1)).

In this paper, we propose the inclusion of acoustic information using our Gaussian classifier.

So, we will add two new features to our score vector: the acoustic score obtained in the phone recognizers of both languages. Again, the approach can be easily extended to several languages.

In our first experiments we observed, as in Section 5.2.3, that the values of the acoustic score were not homogeneous at all, and so, the estimated distributions had a big overlap between the languages that we wanted to classify. All experiments using those scores provided worse results.

Then, we decided to use again the "differential scores" idea: we used the difference between the score for the Spanish phone recognizer and the score for the English phone recognizer as feature value. Again, we observed that the overlap between the estimated distributions reduced drastically. So, we just have one feature in the acoustic score vector.

6.1.1. Database considered

Obviously, we need to estimate the acoustic score distributions using non-training data. So, the dataset chosen for this task is the language models training list, because those sentences have not been used to train the phone models.

One important consideration here is that, in fact, we have trained Gaussian distributions for allophone sequence scores and acoustic scores separately, as they use different lists for the estimation. This is no problem at all, it is very similar to the treatment of different feature vectors in HMM models.

6.1.2. Results

In this point in our experiments, we decided to increase slightly the test set size from 500 sentences up to 700 sentences both in Spanish and English in order to increase the significance of the results. That is why there is a slight change in the results for the Gaussian classifier from Table 8. In Table 9, we can see the results using the Gaussian classifier with two distributions, one for the PPRLM scores and the other one for acoustic scores. As we can see, the improvement is limited for the Minimum (we are very close to a top performance considering that sentences are only 4.5 seconds long in average), but in Average it is remarkable. The average includes experiments giving more relevance to unigram and bigram and, so, results show that acoustic information complements better the least robust systems.

Table 9. Results with acoustic scores

	Average	Minimum
Gaussian classifier	5.42	3.74
+ acoustic scores	4.69 (13.5%)	3.67 (2.0%)

6.2. Multiple-Gaussian classifier

One of the nicest characteristics of a Gaussian classifier is that we can grow up to multiple Gaussians to better model the distribution that represents our classes. Of course, we will need more data to have a reliable estimation of these Gaussians. We will show here that with our data we can estimate reliable multiple-Gaussian distributions using both sources of information: allophone sequence and acoustic score. We have used different number of Gaussians for both of them, as the dimension of the feature vector is completely different: 6 features for sequence score and 1 feature for the acoustic score.

To increase the number of Gaussians we have followed the classical HMM modeling approaches (Gaussian splitting and Lloyd reestimation after each splitting), so we will not describe them here.

In Table 10 we can see a summary of results obtained using different numbers of Gaussians for both scores.

Table 10. Multiple-Gaussian classifier

Number of Gaussians			
LM score	Acoustic score	Average	Minimum
1	1	4.69	3.67
2	1	4.35 (7.2%)	3.52 (4.1%)
2	2	4.14 (11.7%)	3.31 (9.8%)
3	1	**4.01 (14.5%)**	3.24 (11.7%)
3	2	4.12 (12.1%)	3.23 (12.0%)
3	3	4.06 (13.4%)	3.31 (9.8%)
4	1	4.15 (11.5%)	3.38 (7.9%)
4	2	4.20 (10.5%)	**3.16** (13.9%)
4	3	4.08 (13.0%)	3.17 (13.6%)
4	4	4.09 (12.8%)	3.31 (9.8%)

We can extract several interesting conclusions from these results:

- The improvements are really remarkable, up to 14% in minimum value and almost 15% in the average of all experiments.
- As we expected, the best system uses more Gaussians for LM score than for acoustic score.
- It is a nice feature that all systems provide better results than the mono-Gaussian system, showing that there is enough training data for the multiple-Gaussian system.
- There are better improvements in the Average value. Again, the more powerful estimation of multiple Gaussians has more relevance in the less robust systems (the ones with bigger weights for unigram and bigram).
- After all these improvements, the difference between the Average and Minimum values has reduced drastically, showing the robustness of these techniques which reduces the importance of the n-gram weights from equation (1). This is a very nice feature in any PPRLM system.

7. Conclusions

We have described several improvements in a language identification system using PPRLM scores and acoustic information. The system has improved from 6.80% to **3.16%** error rate, which is a remarkable **53.5%** relative improvement. The results are outstanding, as the average duration of the sentences is just 4.5 seconds, although they are difficult to compare with other systems where longer utterances are compared.

Increasing the sentence minimum duration to 2 seconds instead of 0.5 (5.3 seconds average duration) we obtain a **0.82%** error rate. So, most errors in our system come from extremely short sentences.

The most significant improvements have been obtained using the following techniques:

- The application of the variable additive factor in score computation provided a significant error reduction in all cases. It even compensated the bias mismatch in the LM scores, as the results have shown.
- For the classifier, we compared the bias removal technique (up to 19% error reduction) and a Gaussian classifier (up to 37% error reduction), showing that the last one provides better results and has the potential to include additional information. To estimate them, the acoustic models training list can be used with success. The use of differential scores to estimate the Gaussian distributions is also crucial for the technique.
- The inclusion of acoustic score in the Gaussian classifier provided a 2% error reduction and the increase in the number of Gaussians provided an additional 14% error reduction.

8. References

[1] Zissman, M.A., "Comparison of four approaches to automatic language identification of telephone speech," IEEE Trans. Speech and Audio Processing, vol. 4(1), pp. 31-44, 1996.

[2] Córdoba, R., G. Prime, J. Macías-Guarasa, J.M. Montero, J. Ferreiros, J.M. Pardo, "PPRLM Optimization for Language Identification in Air Traffic Control Tasks". Eurospeech 2003, pp. 2685-2688.

[3] Torres-Carrasquillo, P.A., Reynolds, D.A., Deller Jr., J.R., "Language identification using Gaussian mixture model tokenization", IEEE ICASSP 2002, pp. I-757-760.

[4] Torres-Carrasquillo, P.A., Singer, E., Kohler, M.A., Greene, R.J., Reynolds, D.A., Deller Jr., J.R., "Approaches to Language Identification Using Gaussian Mixture Models and Shifted Delta Cepstral Features", ICSLP 2002, pp. 89-92.

[5] Wong, E., Sridharan, S., "Methods to Improve Gaussian Mixture Model Based Language Identification System", ICSLP 2002, pp. 93-96.

[6] Fernández, F., R. de Córdoba, J. Ferreiros, V. Sama, L. F. D'Haro, J. Macías-Guarasa. 2004. "Language Identification Techniques based on Full Recognition in an Air Traffic Control Task". Interspeech-ICSLP, pp. II-1565-1568.

[7] Ma, B., C. Guan, H. Li, C.H. Lee. 2002. "Multilingual Speech Recognition with Language Identification". ICSLP, pp. 505-508.

[8] Navratil, J. 2001. "Spoken Language Recognition – A Step Toward Multilinguality in Speech Processing". IEEE Transactions on Speech and Audio Processing, Vol. 9, Nº 6, Sept. 2001, pp. 678-685.

[9] Gauvain, J.L., A. Messaoudi, H. Schwenk. 2004. "Language Recognition using Phone Lattices". ICSLP, pp. I-25-28.

[10] Ramasubramaniam, V., A.K.V. Sai Jayram, T.V. Sreenivas. 2003. "Language Identification using Parallel Phone Recognition". Workshop on Spoken Language Processing, India.

[11] Jin, Q., Schultz, T., Waibel, A., "Phonetic Speaker Identification", ICSLP 2002, pp. 1345-1348.

[12] INVOCA Project Synopses. Eurocontrol. Analysis of R&D in European Programmes. http://www.eurocontrol.int/ardep-arda/public/jsp/Ardep.jsp?MENU_ITEM=1014&Proj=AEN043.