Language Identification based on n-gram Frequency Ranking

R. Cordoba, L. F. D'Haro, F. Fernandez-Martinez, J. Macias-Guarasa, 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
{cordoba,lfdharo,efhes,macias,jfl}@die.upm.es

Abstract

We present a novel approach for language identification based on a text categorization technique, namely an n-gram frequency ranking. We use a Parallel phone recognizer, the same as in PPRLM, but instead of the language model, we create a ranking with the most frequent n-grams, keeping only a fraction of them. Then we compute the distance between the input sentence ranking and each language ranking, based on the difference in relative positions for each n-gram. The objective of this ranking is to be able to model reliably a longer span than PPRLM, namely 5-gram instead of trigram, because this ranking will need less training data for a reliable estimation. We demonstrate that this approach overcomes PPRLM (6% relative improvement) due to the inclusion of 4-gram and 5-gram in the classifier. We present two alternatives: ranking with absolute values for the number of occurrences and ranking with discriminative values (11% relative improvement).

Index Terms: Language Identification, n-gram frequency ranking, text categorization, PPRLM

1. Introduction

The most used technique in Language identification (LID) is the phone-based approach, like Parallel phone recognition followed by language modeling (PPRLM) [1]-[3], which classifies languages based on the statistical characteristics of the allophone sequences with a very good performance.

An interesting variant of PPRLM is presented in [4] with several proposals: different ways to combine the 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. In [5] they 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 we use.

PPRLM does not model long-span dependencies. As we checked during the work carried out in [2] and [3], best results can be obtained using trigrams, but with 4-grams language models results are slightly worse, probably due to insufficient training data to estimate them reliably.

We thought that one way to overcome this issue and include 4-gram (and even 5-gram) information in our language identification system was to use text categorization techniques based on the ranking of occurrences of each n-gram, as in [6] where the ranking is applied to written text.

As the information used by the classification system is very similar to PPRLM (frequency of occurrence of n-grams), we were afraid that results could be at most similar to PPRLM, but as we will see, due to the contribution of 4-grams and 5-grams, we have been able to overcome PPRLM.

2. System description

2.1. Database

We use a continuous speech database (referred to Invoca database from now on), which consists of very spontaneous conversations between controllers and pilots. It is quite a difficult task, noisy and very spontaneous. 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, and they mix Spanish for greetings and goodbyes even when the rest of the sentence is in English.

For the training set, we had some 8 hours of speech for Spanish and 6 hours for English. For the validation set, we had some 1 hour for both languages and 700 sentences. 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., which is another important complication for the LID task.

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. All models use 10 Gaussians densities per state per stream.

2.3. Brief 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. First, a phone recognizer takes the speech utterance and outputs the sequence of allophones corresponding to it. Then, the sequence of allophones is used as input to a language model (LM) module. In recognition, the LM module scores the probability that the sequence of allophones corresponds to the language. It can use several phone recognizers modeled for different languages. 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 (weights α_1, α_2, and α_3 for unigram, bigram and trigram, respectively). All systems using 4-gram LMs provided worse results.

2.4. Gaussian classifier for LID

As is described in [5], the general PPRLM approach has a flaw: there is the possibility of having a different bias in the log-likelihood score for the languages considered. This is especially relevant when the phone recognizers have a different number of units (we have 49 units for Spanish and

61 for English). The language with fewer units will have higher probabilities in the LM score, and so the classifier will tend to select that language. To tackle this issue, we proposed in [3] to use a Gaussian classifier instead of the usual decision formula applied in PPRLM. 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. 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, using a diagonal covariance matrix, and the distribution which is closer to the input vector is the one selected as identified language.

To estimate the Gaussian distribution we used the acoustic models training list, as this data does not participate in the LM estimation. We demonstrated in [3] that it was a good option in order to make a better use of the training list.

One important conclusion of that work is that, instead of absolute values, we need to use differential scores: the difference between the score obtained by the LM of the same language of the acoustic models considered (Spa-Spa or Eng-Eng) and the score obtained by the other 'competing' language(s): SC0 – SC1 and SC3 – SC2 in Figure 1. So, this score can be computed both in training and testing. We applied it to unigram, bigram and trigram separately, with 6 features in total that are listed in Table 1.

Figure 1. PPRLM Scores

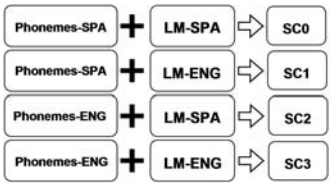

Table 1. Differential score vector

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

One nice feature of a Gaussian classifier is that we can increase the number of Gaussians to better model the distribution that represents our classes and have a Multiple-Gaussian classifier. To increase the number of Gaussians we have followed the classical HMM modeling approaches (Gaussian splitting and Lloyd reestimation after each splitting). More details of our system can be found in [2]and [3]. In Table 2 we can see the results using PPRLM and our Multiple-Gaussian classifier for the optimum combination of weights. We will consider this our baseline.

Table 2. LID results with PPRLM

Gaussians	Error rate (%)
1	3.74
2	3.82
3	3.67
4	3.80
5	**3.60**
6	**3.60**

3. n-gram Frequency Ranking

3.1. Base system: all n-grams in one ranking

We use the same input as PPRLM: the sequence of allophones generated by the phone recognizer. As proposed in [6], we use all training data to compute the number of occurrences of each n-gram (n=1 to 5). We sort those counts by the number of occurrences, and keep only the M most frequent n-grams, which will form the ranking for that input language. It is known ([6]) that the top n-grams are almost always highly correlated to the language.

We will use this ranking instead of the LM module considered in PPRLM. So, we will also have 4 independent rankings, as we had 4 LMs in PPRLM (see Figure 1). As in PPRLM, we estimate these global rankings using the acoustic models training list.

In testing, for each input sentence a ranking is created using the same procedure. Then, the distance between the input sentence ranking and all 4 global rankings is computed. The distance measure is the following (we add the difference in the ranking position for all n-grams in the input sentence):

$$d = \frac{1}{L}\sum_{i=1}^{L} abs(pos\ input_i - pos\ global_i) \quad (1)$$

where L is the number of n-grams in the input sentence. If an n-gram does not appear in the global ranking (meaning that it has not appeared in training or it is not in the top n-grams selected) it is assigned the worst distance: the size of the global ranking. The language identified by the system will be the one with the lowest distance.

In our first approach, we kept the top 400 n-grams, as proposed in [6], but the LID rate was only 7.5% error rate, so we decided to research other alternatives.

One variation from [6] is the application of what we call a "golf score". As the number of occurrences of the n-grams in the input sentence is very low, most n-grams have the same number of occurrences and should have the same position in the ranking. It is the same as a ranking in golf (the sport): all players with the same number of strokes share the same position. It meant a relative improvement of 5%.

Then, we applied our Gaussian classifier to these scores. As we did with PPRLM, we used the differential scores described in Section 2.3. In Table 3, we can see the results varying the ranking size (optimum number of Gaussians): better results are obtained using rankings with 3,000 n-grams.

Table 3. LID results varying the ranking size

Ranking size	Error rate (%)
400	6.40
1000	6.11
2000	5.11
3000	**4.39**
4000	4.42

3.2. n-gram specific rankings

We thought that this global ranking proposed in [6] was not suitable for our task: the top positions were always devoted to the unigrams, bigrams, etc., which we already knew that were less discriminative for language identification. In PPRLM, the optimum result is always obtained with the highest weight for the trigram. So, we decided to have different rankings for each n-gram order (besides that, the procedure is the same).

As the ranking size for unigram and bigram will be different between languages, we need an additional normalization in the distance measure: we divide it by the number of items in the set for that n-gram order.

In our Gaussian classifier we now have 10 features in our vector, the same 6 features from Table 1 for unigram, bigram, and trigram, and 4 new features for 4-gram and 5-gram. In Table 4 we can see the results using this approach. Now, the ranking size presented in the table is the maximum allowed in the ranking creation algorithm, because for unigram and bigram there are less than 2000 different items. There is a nice improvement with this approach. Of course, there is more information as more n-grams are considered globally in the system, but they are reliably estimated. Nevertheless, we are still below PPRLM results (Table 2).

Table 4. LID results with n-gram specific ranking

Ranking size	Error rate	Improv. (%)
1000	4.46	27.0
2000	3.96	22.5
3000	**3.82**	13.0
4000	3.96	10.4

3.3. Measure of separation between distributions

LID experiments can be very time consuming, as we can modify the weights applied for each n-gram. In PPRLM, we just considered up to trigrams, but with this approach we were confident that we could use up to 5-grams, so the combination of weights is huge. Therefore, we decided, to restrict the weights considered in the experiments using, for each feature, information regarding the separation between the pdf distributions for each candidate language. We apply the following formula which is used in feature selection algorithms to reduce the dimensionality:

$$\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2\ \sigma_2^2} \qquad (2)$$

where μ_1 and μ_2 are the mean values for the feature considering Spanish and English input sentences respectively, and σ_1 and σ_2 are the respective covariances. A high value in this formula means that the feature is especially discriminative between the languages. We have always found that there is a very strong correlation among this measure of separation between the Gaussian distributions and the final results in LID (as could be expected, obviously).

In Table 5 we can see the separation which is obtained with PPRLM and n-gram ranking for each n-gram considered (average of the 2 values for each n-gram). So, the discriminative power of PPRLM is higher, especially for the trigram, but the nice thing of n-gram ranking is that we also obtain a nice discrimination with the 4-gram and 5-gram that cannot be used in PPRLM due to insufficient training data. This confirms the results from Table 4, where PPRLM beats our ranking proposal.

Table 5. Comparison of feature discrimination

	PPRLM	n-gram rank.
trigram	10.85	8.42
4-gram	-	6.41
bigram	8.42	5.35
5-gram	-	4.43
unigram	3.12	2.06

4. n-gram discriminative ranking

We considered another solution. As the concept of differential scores always worked well in our system, we thought that we should introduce the same concept in the ranking creation process. We wanted to give more relevance in the ranking (higher positions) to the items that are actually more specific to the language that is being identified, i.e. n-grams that appear a lot for one language but appear very little, or never, in the competing languages. So, we decided to introduce document/topic classification techniques.

We first thought of tf-idf, which is used for topic classification among other things, but as we only have two languages, it only discriminates n-grams that appear in one language but not in the other, and very few n-grams fulfill that. So, we propose a variation of tf-idf, which we describe now. After the original global rankings are created, we have the number of occurrences of each n-gram:

n_1 = occurrences of item *i* in the current language
n_2 = occurrences of item *i* in the competing language (it would be the average in the competing languages to extend this measure to multiple languages)

As the number of total occurrences will be different for each language and n-gram order, before the subtraction a normalization is needed to have comparable amounts. Being N_1 the sum of all occurrences for the current language and N_2 for the competing language(s):

$$n_1' = (n_1 * N_2) / (N_1+N_2)$$
$$n_2' = (n_2 * N_1) / (N_1+N_2)$$

using these normalized values we considered several alternative formulas with the same philosophy as tf-idf for the final number of occurrences considered for the ranking (which we will call n_1'') and studied the separation between the Gaussian distributions for each language obtained using each formula before diving into the LID experiments (see Table 6). To summarize, only the average separation for all 5 n-grams is presented. First, a purely discriminative solution:

$$n_1'' = (n_1' - n_2') / (n_1' + n_2')$$

There is a nice improvement over the non-discriminative ranking, but we tested other alternatives, as including an item frequency term in the formula:

$$n_1'' = n_1' * (n_1' - n_2') / (n_1' + n_2')$$

but with this solution we lost part of the discriminative power. So, we decided to reduce the effect of the first term by taking its logarithm or the square root with nice improvements. The last formula in Table 6 provides the best classification power, probably because it normalizes the values between 1 and -1: 1 meaning that the n-gram appears in the current language but not in the other competing ones ($n_2'=0$), indicating that it is especially relevant for that language; -1 meaning just the opposite ($n_1'=0$), so the n-gram does not appear in the current language.

Table 6. Average feature discrimination (several formulas)

Formula	Av. separation
Original – no discriminative	6.15
$n_1'' = (n_1' - n_2') / (n_1' + n_2')$	6.75
$n_1'' = n_1' * (n_1' - n_2') / (n_1' + n_2')$	6.48
$n_1'' = \log(n_1') * (n_1' - n_2') / (n_1' + n_2')$	6.82
$n_1'' = \mathrm{sqrt}(n_1') * (n_1' - n_2') / (n_1' + n_2')$	7.01
$n_1'' = n_1' * (n_1' - n_2') / (n_1' + n_2')^2$	**7.13**

We can see in Table 7 that the discrimination for the ranking trigram is very similar to the PPRLM trigram, but now we can use 4-grams and 5-grams. Also, results are better than those obtained in Table 4 (non-discriminative ranking).

Table 7. Comparison of feature discrimination

	PPRLM	Discr. ranking
trigram	10.85	10.12
4-gram	-	6.70
bigram	8.42	7.24
5-gram	-	4.43
unigram	3.12	2.06

4.1. Threshold

One factor that has to be also addressed with these measures is that they are very prone to overtraining: n-grams that just appear once or twice in training for one language and never for the competing language(s), will be at the top position of the list, even though they are probably irrelevant.

So, we decided to apply a threshold: if (n_1''+ n_2'') < *threshold*, send the item to the last position in the ranking. After some testing, the optimum threshold was: 6-4-3-2-2 for unigram, bigram, trigram, 4-gram, and 5-gram respectively.

4.2. LID results

In Table 8 (3rd column) we can see the LID results using this technique (in parenthesis the relative improvement). For simplicity, we only present the results for a ranking size equal to 3000 (the provided). We can see that, even with one Gaussian, results are better. Probably, the reason is that we now have a 10 feature vector instead of 6 with PPRLM, so it is more difficult to estimate reliably several Gaussians with our training database. The improvement over PPRLM for the best results is **6.1%** (3.38 versus 3.60). Over the non-discriminative ranking, it is **11.5%** (3.38 versus 3.82).

Table 8. LID: PPRLM versus discriminative ranking

Gaussians	PPRLM	Discrim.	Discrim + acoustic
1	3.74	**3.38** (9.6%)	2.95 (12.7%)
2	3.82	3.46 (9.4%)	3.09 (10.7%)
3	3.67	3.60 (1.9%)	3.09 (14.2%)
4	3.80	3.46 (8.9%)	3.02 (12.7%)
5	3.60	3.60 (0.0%)	3.02 (20.7%)
6	3.60	**3.38** (6.1%)	**2.73** (**19.2%**)

4.3. Fusion with PPRLM and acoustic scores

Although it is not the objective of this paper, as we proved in [3] that the fusion of PPRLM and acoustic scores provided better results using different feature vectors in our Gaussian classifier, we have checked that indeed the fusion of this n-gram discriminative ranking with acoustic scores also improved the system. As we can see in Table 8 (4th column), the results are also outstanding, obtaining even better results than the fusion of PPRLM + acoustic scores, which provided slightly smaller improvements (around 10-14%).

The fusion of PPRLM with this technique provides smaller improvements: an average (all Gaussians) of 8.2%, with a best result of **3.24%** error rate. In any case, this is even surprising, as they use the same source of information.

The best score of the fusion of all 3 (PPRLM + Discrim. ranking + acoustic) is **2.59**%, which is a nice additional improvement (5%) over "Discrim. ranking + acoustic".

4.4. Longer span of the technique

We also checked the relevance of 4-grams and 5-grams in LID with this technique. In Table 9 we can see that the LID results considering only up to 4-gram or up to trigram are worse than using all n-grams. So, we are clearly taking advantage of this longer span using this technique.

Table 9. Independent ranking for each n-gram

	Best result
All n-grams	3.38
Up to 4-gram	3.82
Up to trigram	4.13

5. Conclusions

We have demonstrated that the n-gram Frequency Ranking approach can overcome PPRLM thanks to the longer span that can be modeled. To obtain this improvement, the following issues have been crucial:

- The ranking size should be 3,000.
- n-gram specific rankings should be used. A common ranking for all n-grams is clearly a worse solution.
- The measure of separation between pdf distributions (Section 3.3) is a good tool to anticipate which features are going to be actually discriminative for the LID task.
- n-gram discriminative rankings with the normalized value for the number of occurrences are able to overcome PPRLM (6.1% relative improvement).
- The fusion with acoustic scores (19% improvement) and with PPRLM (7.2%) provides the best results.

This approach can be easily extended to multiple languages just averaging the number of occurrences for competing languages, as we describe in Section 4. As future work, we will check these results with a bigger database.

6. Acknowledgements

This work has been partially funded by the Spanish Ministry of Education & Science under contracts DPI2004-07908-C02-02 (ROBINT) and TIN2005-08660-C04-04 (EDECAN-UPM) and by UPM-CAM under contract CCG06-UPM/CAM-516 (ATINA).

7. References

[1] Zissman, M.A., "Comparison of four approaches to automatic language identification of telephone speech," IEEE Trans. Speech&Audio Proc., v. 4, 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] Córdoba, R., et al. "Integration of acoustic information and PPRLM scores in a multiple-Gaussian classifier for Language Identification". IEEE Odyssey 2006.

[4] Navratil, J. 2001. "Spoken Language Recognition – A Step Toward Multilinguality in Speech Processing". IEEE Trans. Speech&Audio Proc., Vol. 9, pp. 678-685.

[5] Ramasubramaniam, V., et al. 2003. "Language Identification using Parallel Phone Recognition". Workshop on Spoken Language Processing, India.

[6] Cavnar, W. B. and Trenkle, J. M., "N-Gram-Based Text Categorization", Proc. 3rd Symposium on Document Analysis & Information Retrieval, pp. 161-175, 1994.