# ITERATIVE SPEAKER ADAPTATION USING MLLR

*José R. Navarro Cerdán*[1], *Carlos Martínez Hinarejos*[2],
*Antonio L. Lagarda Arroyo*[2], *Luis Rodríguez Ruiz*[1]

[1]Instituto Tecnológico de Informática
[2]Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia, Cno. Vera s/n, 46022, Valencia, Spain
jonacer@iti.upv.es,cmartine@dsic.upv.es,alagarda@dsic.upv.es,lrodrig@iti.upv.es

## ABSTRACT

Speech recognition systems are usually speaker-independent, but they are not as good as speaker-dependent systems for specific speakers. An initial speaker-independent system can be adapted to improve recognition accuracy by transforming it into a speaker-dependent system. In this work, a new general acoustic model adaptation technology is presented, using the MLLR algorithm iteratively in a supervised manner. Experiments have been performed on the TT2 Spanish speech corpus. The initial acoustic models were trained from the Albayzin speech database. Their results, which were obtained for 10 speakers, show an improvement in speech recognition accuracy.

## 1. INTRODUCTION

Speech recognition improvements have contributed to the widespread use of speech recognition systems in several applications [1, 2, 3]. Speech recognition systems rely on acoustic and language models to perform the recognition of input utterances. This work deals with only one part of speech recognition system, acoustic model. They model sequences of feature vectors that describe a specific sound (phonemes, syllabes, etc.). They are usually continuous-density Hidden Markov Models (HMM) [4, 5], in which each state models its output distribution using a mixture of Gaussians. Each Gaussian is defined by a feature mean vector and a covariance matrix.

Parameter estimation of acoustic models is done by means of the well-known Baum-Welch algorithm [6]. A good estimation of these models requires a lot of training data. This makes speaker-independent systems common because obtaining a large amount of training data for systems of this kind is easier than obtaining a large amount of data for speaker-dependent systems. However, for a specific speaker, more accurate results can be achieved by using speaker-dependent acoustic models, provided that sufficient data is available. Unfortunately, obtaining enough speaker-specific data for speaker-dependent acoustic model estimation is very difficult.

The solution consists of obtaining a speaker-dependent acoustic model by adapting a speaker-independent acoustic model to a specific speaker, using only a small amount of specific-speaker data.

## 2. TYPES OF ADAPTATION TECHNIQUES

Several speaker adaptation techniques have been developed in the last few years [7, 8, 9]. These techniques can be divided into two main groups, depending on what is modified (*the input signal* or *the acoustic model*). The most important speaker adaptation techniques are described in [10, 11].

### 2.1. Spectral mapping techniques

In these techniques, modifications are made on the input signal. The acoustic signal (or its codification) is altered to adapt the signal to a general acoustic model; therefore, with these techniques, the signal of the new speaker is closer to the signal of the reference speakers. The main techniques are:

- *Dynamic Time Warping (DTW)*[12]: Dynamic time-warping is a dynamic programming algorithm that finds the reference signal alignment that minimizes the distance to input signal.
- *Spectral-Bias*[13]: This method uses the information incorporated in speaker-independent Hidden Markov Models (HMM) and estimates a transformation of the means of the models. Although this method transforms the means of the HMM, it is included in this group because the goal of the method is to improve the match between the reference speakers and the new speaker (the *spectral mapping* idea) rather than to improve the modeling accuracy for the new speaker (the *model mapping* idea).
- *Vocal Tract Length Normalizacion (VTLN)*[14]: Human vocal tract length produces variation in the main components of the source speech signal. Assuming

This work has been partially supported by Spanish CICYT under project TIC2003-08681-C02-02 and by the European Union under grant IST-2001-32091

a different vocal tract length for each individual speaker, source speech signal from a speaker can be transformed into a normalized signal using the VTLN algorithm. This signal can then be used to train the acoustic models. This technique uses a simple transformation function that depends on a parameter (warping factor) and the signal frequency in each instant.

### 2.2. Model mapping techniques

In these techniques, modifications are made on the acoustic models. In this case, acoustic models are altered in order to make them closer to the source input signal from the speaker. In other words, we approximate general acoustic models to the input signal from the speaker. The main techniques are:

- *Maximum adaptation a posteriori (MAP)*[8]: This is a general probability distribution estimation technique that allows previous knowledge to be introduced in the estimation process. In this case, the knowledge is the parameters of the speaker independent acoustic models.
- *Regression-based Model Prediction, (RMP)*[15]: This method is based on linear regression. The idea consists of using the available adaptation material to make an initial *maximum a posteriori (MAP)* adaptation for the model means. These *MAP* estimates are then used to predict the means of the models which were not present in the adaptation data; this is done via a set of regression coefficients, which are computed using previously trained speaker-dependent models.
- *Maximum Likelihood Linear Regression, (MLLR)*[7]: With this method, the feature means of general acoustic models are adapted to the speaker's voice using a linear regression model that is estimated by means of maximum likelihood. This is the method that we used for our speaker adaptation system and is explained below in Section 3.

In this article, the results obtained with the iterative application of *Maximum Likelihood Linear Regression Model* in a supervised fashion are presented. Our technique is based on making successive speaker adaptations by means of the MLLR algorithm, to improve the speaker's acoustic models reusing the adaptation data. The final goal is to obtain better accuracy in speech recognition for that specific speaker.

## 3. THE MLLR SPEAKER ADAPTATION TECHNIQUE

This method is based on the application of an adaptation matrix, *W*, on the acoustic model parameters. This matrix *W* is computed by means of maximum likelihood, with the general acoustic model parameters without adaptation and the speaker voice to be adapted as input data. An acoustic model that is completely adapted to the speaker is obtained by applying this method.

To start with the MLLR algorithm description [7, 11], we must consider the case of a continuous density HMM system with Gaussian output distributions. A particular distribution, $s$, will be characterised by a mean vector, $\mu_s$, and a covariance matrix $C_s$. Given a parametrized speech frame vector $o$, the probability density of that vector being generated by distribution $s$ will be $b_s(o)$

$$b_s(o) = \frac{1}{(2\pi)^{n/2}} e^{-1/2(o-\mu_s)' C_s^{-1} (o-\mu_s)}$$

where $n$ is the dimension of the observation vector and $'$ denotes the transpose vector.

The MLLR algorithm can be summarized as follows: for each Gaussian $s$, compute the new speaker estimated $\hat{\mu}_s$ parameter from the general $\mu_s$ parameter. This is obtained using:

$$\hat{\mu}_s = W_s \xi_s$$

where:

- $W_s$ is the adaptation matrix.
- $\xi_s = [\omega, \mu_{s_1}, \ldots, \mu_{s_n}]$ is the extended vector of means, with shift $\omega$.

This computation can also be done for the covariance matrix, but this additional adaptation does not usually provide better results [11]. Thus, the probability density function for the adapted system for the Gaussian $s$ is:

$$b_s(o) = \frac{1}{(2\pi)^{n/2}} e^{-1/2(o-W_s\xi_s)' C_s^{-1} (o-W_s\xi_s)}$$

Since it is usually not possible to estimate $W_s$ for each Gaussian $s$, *regression classes* are defined as Gaussian sets that share the same adaptation matrix.

The number and optimum composition of *regression classes* cannot be defined analytically. Thus, its selection is usually based on the amount of available adaptation data, the phonetic split between models (decision trees), and the different join critera between models (phonetic features, distance between models, etc).

To estimate the transformation matrix, given a *regression class* $R = \{s_1 k, s_2 k, \ldots, s_R k\}$, $W_s$ is estimated by maximum likelihood:

$$\hat{W}_s = \max_{W_s} \Pr(O_p|\hat{\lambda})$$

where:

- $O_p$ is the sequence of observations.

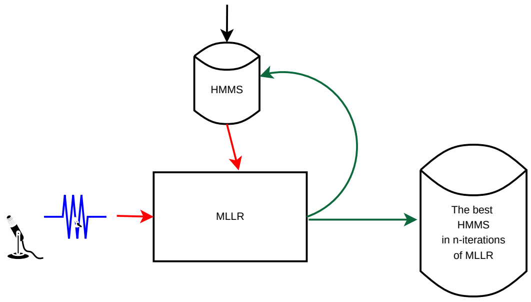

**Figure 1**. Iterative MLLR architecture.

- $\hat{\lambda}$ is the model obtained applying $W_s$.
- $\hat{W}_s$ is the optimum estimation of the adaptation methods.

$\hat{W}_s$ is obtained with the optimization of an auxiliar function, $Q$.

$$Q(\lambda, \hat{\lambda}) = \sum_{\theta \in \Theta} \sum_{k \in \Omega_b} \Pr(O_p, \theta, k|\lambda) \log(\Pr(O_p, \theta, k|\hat{\lambda}))$$

where,

- $\Theta$ is the state sequence set.
- $\Omega_b$ is the Gaussian set.

The estimation of $\hat{W}_s$ with this formulation is usually complicated and time-consuming. To estimate it, a Viterbi approximation can be used, which corresponds to the following formula:

$$\hat{W}_s = \left( \sum_{t=1}^{T} O^t \xi'_{s_r k} \right) \left( \sum_{t=1}^{T} \xi_{s_r k} \xi'_{s_r k} \right)^{-1} \quad (1)$$

To apply this approach, the observations $O^t$ are initially decoded in a forced way (using a transcription). The decoding process gives the Gaussian with maximum likelihood for each observation, whose mean is $\xi_{s_r k}$. With this data, Equation (1) is applied to compute $\hat{W}_s$.

## 4. ADAPTATION ARCHITECTURE

This paper provides new results for speaker adaptation using the MLLR algorithm iteratively instead of only once. The results show that successive adaptations of adapted models can significantly improve the results.

Figure 1 shows the architecture of the method. Initially, the input data are simple wave files of sentences, that have been transcribed, obtained from the speaker to be adapted; they are the initial source of information. These wave files are passed to the MLLR algorithm along with the general acoustic model to be adapted. MLLR gives a new acoustic model that is adapted to the speaker.

This classic application of the MLLR algorithm can be improved by readapting the first adapted acoustic model using the same method. In a second iteration of the algorithm, the first adapted acoustic model acts as the new general acoustic model, which is then adapted using the MLLR algorithm (see the feed-back in Figure 1). Thus, the MLLR algorithm uses the same wave files, that were transcribed initially. In the following iterations, the same wave files and their transcriptions (which are supervised) are used to reestimate a new acoustic model substituting the original acoustic model with the new adapted model. If several iterations are done and the sentence accuracy rate (SAR) of a test set of the adapted speaker for each adaptation is obtained, it is possible to determine whether the results have been improved. This will also obtain the best acoustic model.

Our initial set of acoustic models was obtained from the Albayzin Spanish speech corpus [16]. The acoustic models were Hidden Markov Models (HMM) that represented monophones. Their topology is the classical three-state, left-to-right with loops and without skips. The output distribution for each state was modelled by a mixture of 128 Gaussians with diagonal covariance matrixes. The number of components of the Gaussians was 33 (ten cepstrals plus energy, plus first derivative and acceleration).

## 5. CORPUS DESCRIPTION

The TT2 project [17] is devoted to the construction of Computer Aided Translation (CAT) systems. In this project, text translation is combined with speech input to improve the performance of the human translator. The usual scenario of an interaction between the computer application and the human translator follows these steps:

1. The computer application proposes a translation of the current sentence.
2. The human translator accepts part (a prefix) of the proposed translation.
3. The human translator types in possible corrections.
4. The computer application dynamically changes its proposed translation as the human translator types in the corrections.
5. The human translator returns to step 2 until the current sentence is completely translated.

There are several ways in which the human translator can accept a prefix. In the classical approach, s/he points with the mouse at (or uses the keyboard to move to) the end of the correct part of the sentence. In the case of speech input, s/he can utter any subsentence (one or more words) that is present in the translation, and which is perhaps preceded by the words “accept” and/or “until”. The prefix up to that subsentence will be accepted. In case

**Table 1**. Some examples of uttered subsentences and selected prefixes for the proposed sentence “adición de fuentes a la lista de recursos”.

| Uttered sentence | Selected prefix |
|---|---|
| *lista* | *adición de fuentes a la lista* |
| *aceptar hasta fuentes* | *adición de fuentes* |
| *hasta de* | *adición de* |
| *hasta de recursos* | *adición de fuentes a la lista de recursos* |

of ambiguity, the accepted prefix will be the shortest one. Some examples are presented in Table 1.

A speech corpus was acquired to simulate this scenario when translating Xerox printer manuals to Spanish (Xerox corpus)[17]. This acoustic corpus consisted of a total of 7,489[1] utterances of subsentences derived from the sentences of the task. These subsentences were utterances of 125 complete sentences of the task, which were chosen from the Xerox corpus. Five segmentations into prefixes and suffixes were randomly performed on this set. A random prefix was selected for each suffix generated. The words “aceptar”, “hasta”, and “aceptar hasta” were added as prefixes to some of these selected subsentences, giving a total number of 625 different sentences to be uttered. A sample of possible segmentations for a sentence is presented in Table 2.

Ten speakers (six male, four female) were recruited. The sentences were divided into five different sets of 125 sentences. One of these sets was chosen as the adaptation set and was common to all the speakers; the other four sets were distributed among the speakers (five speakers shared two of these sets and the other five shared the remaining two sets). The acquisition was performed by each speaker in three different sessions (at different times of the day in order to capture variabilities in speech intonation). Different subsets of the groups were acquired in each session. In each session, the selected speaker uttered a total of 250 utterances (i.e., the selected sentences were repeated twice). Thus, this acquisition gave a total number of 750 utterances per speaker. Of these, 250 sentences were selected to be used for speaker adaptation, and the other 500 sentences were selected to test the system. Both groups of sentences were disjoint.

The acquisition was performed using a high quality microphone, at 16kHz sampling rate and 16 bits per sample. The total duration of the acquired signal was nearly 5 hours, although nearly half of the acquired signal was silence (because of the small length of the uttered sentences). The adaptation material was close to 1.5 hours, given by a total of 2,479 utterances.

## 6. RESULTS

Figures 2, 3 and 4 show the different results for a sample of three speakers. Each figure represents one speaker;

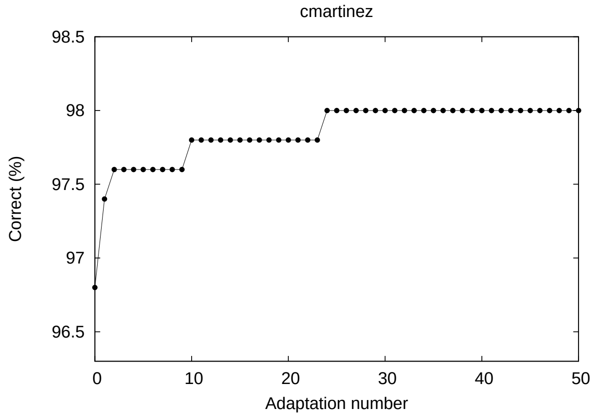

**Figure 2**. Sentence accuracy rate in each iterative adaptation for *cmartinez* speaker.

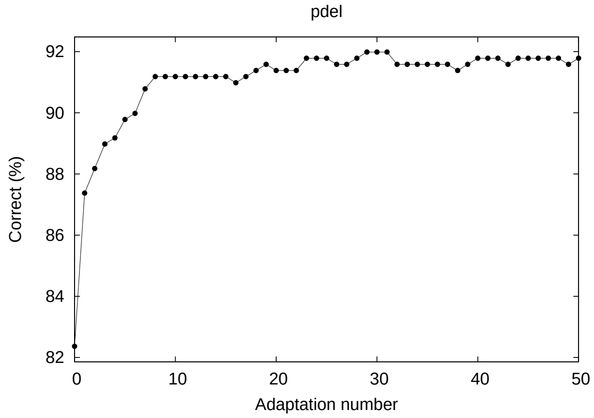

**Figure 3**. Sentence accuracy rate in each iterative adaptation for *pdel* speaker.

the Y-axis represents the sentence accuracy rate (SAR), and the X-axis represents the number of iterations of the MLLR algorithm (up to a total of 50 iterations). Thus, $x = 0$ means SAR with the general acoustic models; $x = 1$ means SAR with one iteration of the MLLR algorithm; $x = 2$ means SAR with two iterations of the MLLR algorithm, and so on. The *cmartinez* speaker in (Figure 2) shows how iterative adaptations progressively improved SAR results. Two more speakers (i.e., three out of ten) present a similar behaviour.

The *pdel* speaker in Figure 3 shows irregular improvement with the different adaptations (i.e., sometimes one more iteration improved the results and sometimes it made them worse). This irregular behaviour also appeared in other five speakers.

The *nalcacer* speaker in Figure 4 shows that there was no improvement with any adaptation. The best result was obtained with the initial general acoustic models without adaptation. No other speaker presented a similar evolution in the results.

[1] The original number of sentences was 7,500, but some of them were corrupted.

**Table 2**. Example of prefixes, suffixes, and prefixes of suffixes randomly derived for the sentence "adición de fuentes a la lista de recursos".

| Prefix | Suffix | Prefixes of the suffix |
|---:|:---|:---:|
| *adición de* | *fuentes a la lista de recursos* | *fuentes a, fuentes a la lista de recursos* |
| *adición de fuentes* | *a la lista de recursos* | *a la lista de* |
| *adición de fuentes a* | *la lista de recursos* | *la, la lista de* |
| *adición de fuentes a la lista* | *de recursos* | *de recursos* |
| *adición de fuentes a la lista de* | *recursos* | *recursos* |

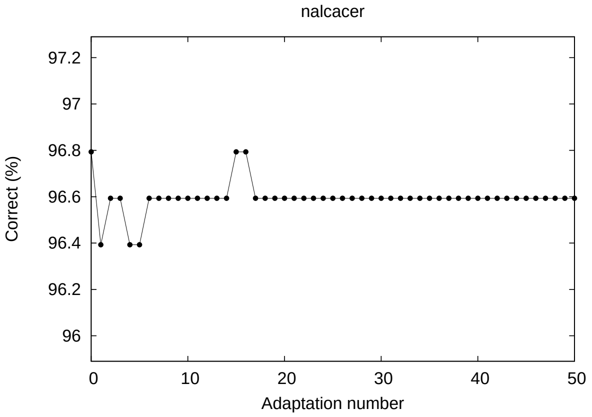

**Figure 4**. Sentence accuracy rate in each iterative adaptation for *nalcacer* speaker.

In Figure 5 it is shown a comparison among the before-mentioned three speaker prototypes, and the mean behaviour of the ten speakers in the study.

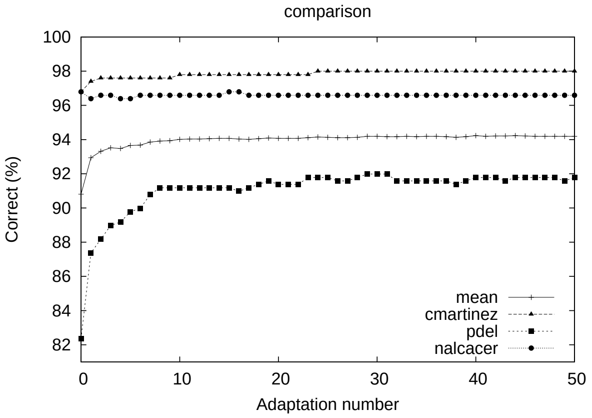

**Figure 5**. Iterative adaptation comparison of sentence accuracy rate for 3 prototype speakers and the mean of 10 speakers in the complet study.

**Table 3**. Speaker sentence accuracy rate.

| Speaker | No adapt. | 1 adapt. | Best adapt. | Iteration |
|---|---|---|---|---|
| alagarda | 86.32 | 90.74 | **93.16** | 29 |
| ecubel | 91.37 | 94.71 | **96.08** | 24 |
| cmartinez | 96.80 | 97.40 | **98.0** | 24 |
| jcivera | 82.91 | 86.64 | **90.18** | 11 |
| jandreu | 94.20 | 95.80 | **96.80** | 40 |
| pdel | 82.36 | 87.37 | **91.98** | 23 |
| evidal | 89.16 | 90.36 | **90.56** | 3 |
| lrodriguez | 96.20 | 97.20 | **98.00** | 4 |
| mnacher | 91.97 | **92.77** | 92.77 | 1 |
| nalcacer | **96.79** | 96.39 | None | 0 |

**Table 4**. Sentence accuracy rate means.

| | |
|---|---|
| **Mean before adaptation** | 90.81 |
| **Mean at first adaptation** | 92.94 |
| **Mean with best adaptation** | 94.43 |

The first three columns of Table 3 represent the SAR results for the ten speakers: column 1, without adaptation; column 2, with only one adaptation; and column 3, with the best adaptation. The last column represents the number of iterations of the adaptation algorithm that were used to obtain the best result from a total of 50 iterations.

From these results, it seems clear that in most cases, iterative adaptation provides a significant improvement in recognition accuracy. What is not clear is the optimal number of iterations of MLLR that must be applied; most speakers need more than 20 iterations, but others get optimal results with less than 5 iterations. In only one case did the application of the adaptation make the results worse than those obtained with non-adapted models.

Table 4 shows the mean SAR results: without adaptation; with only one adaptation; and, finally, with the best adaptation for each speaker, from a set of 50 iterative adaptations. These mean results, demonstrate that, in general, iterative adaptation improves recognition accuracy.

## 7. CONCLUSIONS AND FUTURE WORK

The main conclusion is that, in general, several iterative adaptations seem to improve the speech recognition accuracy. However, with this technique, it is difficult to know what the optimal number of adaptations is because sometimes more adaptations can make the system results worse.

In the future, we plan to formalise this new adaptation technique mathematically. One interesting point is to obtain an automatic and test-independent way of determining the optimal number of iterations. On the practical side, we plan to use this new technique in some industrial speech projects to make the speech recognition systems more reliable.

## 8. REFERENCES

[1] J. Chu-Carroll and R. Carpenter, "Vector-based natural language call-routing," *Computational Linguistics*, vol. 25, no. 3, pp. 361–388, 1997.

[2] A. L. Gorin, G. Riccardi, and J. H. Wright, "How may I help you?," *Speech Communication*, vol. 23, pp. 113–127, 1997.

[3] F. Casacuberta, C. Martínez, F. Nevado, and E. Vidal, "Implementation of an automatic voice-driven telephone exchange," *In Proceedings of the IX Spanish Symposium on Pattern Recognition and Image Analysis*, vol. 25, no. 3, pp. 307–314, May 2001, Benicàssim, Spain.

[4] Rabiner L. and Juang B., "Fundamentals of Speech Recognition," *Prentice Hall*, 1993.

[5] Jelinek F., "Statistical Methods for Speech Recognition," *MIT Press*, 1998.

[6] L. E. Baum., "An inequality and associated maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Inequalities*, , no. 3, pp. 1–8, 1972.

[7] C. J. Leggeter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density Hidden Markov Models," *Computer Speech and Language*, vol. 9, pp. 171–185, 1995.

[8] J. L. Gauvain and C.H. Lee, "Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, April 1994.

[9] R. Kuhn, P. Nguyen, J. C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, and M.Contolini, "Eigen-voices for speaker adaptation," *In Proceedings of the International Conference on Speech and Language Processing, ICSLP*, vol. 5, pp. 1771–1774, 1998, Sidney.

[10] Carlos Martínez Hinarejos, "Seminario de Técnicas de adaptación al locutor," 2006, DSIC-UPV, Valencia, España.

[11] Heidi Christensen, "Speaker Adaptation of Hidden Markov Models using Maximum Likelihood Linear Regression," *Master Thesis*, 1996, Aalborg University.

[12] Joseph B. Kruskal and Mark Liberman, "The symmetric time-warping problem: from continuous to discrete," in *Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983, Massachusetts.

[13] S. J. Cox and J. S. Bridle, "Unsupervised Speaker Adaptation by Probabilistic Spectrum Fitting," *Proceedings ICASSP-89*, pp. 294–297, 1989, Glasgow.

[14] Li Lee and Richard C. Rose, "Speaker normalization using efficient frequency warping procedures," *In Proceedings of the ICASSP-96*, vol. 1, pp. 353–356, 1996, Atlanta, GA.

[15] S. M. Ahadi and P. C. Woodland, "Rapid Speaker Adaptation Using Model Prediction," *In Proceedings of the ICASSP-95*, pp. 684–687, 1995, Michigan.

[16] J.E. Díaz-Verdejo, A. M. Peinado, A. J. Rubio, E. Segarra, N. Prieto, and F. Casacuberta, "Albayzin: a task-oriented Spanish speech corpus," *In Proceedings of First Intern. Conf. on Language Resources and Evaluation (LREC-98)*, vol. 1, pp. 497–501, 1998.

[17] SchlumbergerSema S.A., Instituto Tecnológico de Informática, Rheinish Westfälische Technische Hochschule Aachen Lehrstul für Informatik VI, Recherche Appliquée en Linguistique Informatique, Laboratory University of Montreal, Celer Soluciones, Société Gamma, and Xerox Research Centre Europe., "TT2. TransType2 - Computer Assisted Translation. Project Technical Annex," *Information Society Technologies (IST) Programme*, 2001, IST-2001-32091.