Doctoral School on Engineering Sciences
Università Politecnica delle Marche

Extended summary

Development of a Distributed Speech Recognition System

Curriculum: Ingegneria Elettronica, Elettrotecnica e delle Telecomunicazioni

Author

Massimo Mercuri

Tutor

Prof. Claudio Turchetti

Date: 30-01-2013

Abstract. Over the years various devices have been used for the interaction man-machine: mouse, keyboard, joystick and tablet. The speech, in particular, represents a mode of interaction extremely natural to man, seen that represents the method of communication preferential used every day. For this reason, speech technologies have played, and are of particular interest within the scientific community. The speech recognition systems represent the component through which the speech interaction with a system can take place. In particular, in this work, the attention has been placed on the development of a distributed speech recognition system. In this type of systems, the architecture of the speech recognition process is due to a pattern of client-server. The client hosts the front-end which calculates and extracts the representative speech parameters (features). The recognition of the message is carried by the back-end server resident and is based on the processing of the feature stream. It was initially developed a front-end according to standard ETSI ES 202-212, able to work in real time. The front-end was first implemented on the Java platform, and then taken to mobile device, smartphone, based on Android platform. The remote back-end has been developed using an open source toolkit (Sphinx-4) developed by Carnegie Mellon University. Based on the problems identified in the first phase and especially of room for improvement glimpsed, research has subsequently focused on the speaker adaptation techniques. After careful analysis of the literature, in particular as regards the approaches of "speaker adaptation", has been proposed and developed an original algorithm for speaker adaptation, to be applied on the client side. This algorithm is characterized by reduced complexity,

while ensuring high performance. The results obtained note the effectiveness of the proposed method and the functionality of the speech recognition system distributed as a whole.

Keywords. Automatic speech recognition, Distributed speech recognition, Noise reduction, Speaker adaptation.

1 Problem statement and objectives

The performance of speech recognition systems receiving speech that has been transmitted over mobile channels can be significantly degraded when compared to using an unmodified signal. The degradations are as a result of both the low bit rate speech coding and channel transmission errors.

A Distributed Speech Recognition (DSR) system overcomes these problems by eliminating the speech channel and instead using an error protected data channel to send a parameterized representation of the speech, which is suitable for recognition. The processing is distributed between the terminal and the network. The terminal performs the feature parameter extraction, or the front-end of the speech recognition system. These features are transmitted over a data channel to a remote "back-end" recognizer. The end result is that the degradation in performance due to transcoding on the voice channel is removed and channel invariability is achieved. Based on the problems identified in the first phase and margin improvement glimpsed, research has focused on the speaker adaptation.

Speaker adaptation techniques have proven to be very effective in modern speech recognition systems [1], especially when there are significant mismatches between the training and decoding conditions. In these techniques one starts with a speaker-independent (SI) model, and then tries to accommodate the model to a new speaker to obtain a speaker-dependent (SD) model, using a relatively small amount of speech data from the new speaker. The basic idea is to compensate for the mismatch between training and test conditions by modifying the model parameters on the basis of some adaptation data. Among these techniques the maximum-likelihood linear regression (MLLR) [2] and constrained MLLR (CMLLR) [3],[4] are powerful and widely used methods for speaker adaptation in large-vocabulary continuous speech recognition (LVCSR). MLLR uses the expectation-maximization (EM) criterion to estimate a linear transformation to adapt Gaussian parameters, i.e. their mean and variance, of hidden Markov models (HMMs) [5]. Although the two transformations are estimated separately, the computational complexity is reasonably high. An alternative scheme to adapt both mean vectors and covariance matrices is to use a CMLLR approach,
in which the transformation applied to the covariance matrix corresponds to the transformation applied to the mean vector. It can be shown that CMLLR is equivalent to a transformation in the feature domain. This property makes CMLLR particularly suitable in a distributed speech recognition (DSR) scheme, in which the recognition process is split up into a front-end on the client side primarily related to feature extraction, and a back-end on the server side devoted to the recognition itself.

The main drawbacks of the CMLLR approach are:

- the algorithm is more complex than MLLR;
- it is an iterative process which converges usually after about 30 iterations, but in some cases it does not converge even after 100 iterations [6].

Thus, as the algorithm for the implementation of CMLLR is more complex than standard MLLR, there is a need for simpler algorithms to be efficiently implemented on the client side of a DSR scheme. Then it was proposed an algorithm that meets this requirement, and due to simpler formulation is able to overcome some of the limitations of the CMLLR. It's

worth noting that the iterative CMLLR has exactly the same formulation as the MLLR algorithm, while requiring less iterations than CMLLR to converge. The algorithm has been evaluated by extensive experimentation using the CMU Sphinx4 recognizer in a setting defined for LVCSR and performance comparison with MLLR and CMLLR techniques shows the effectiveness of the approach.

2 Research planning and activities

In the first phase of the research has been implemented a distributed speech recognition system. Then standard for a front-end that ensure compatibility between the terminal and the remote recognizer is the ETSI standar. The ETSI standard DSR front-end ES 202 212 is based on the Mel-Cepstrum representation that has been used extensively in speech recognition systems.

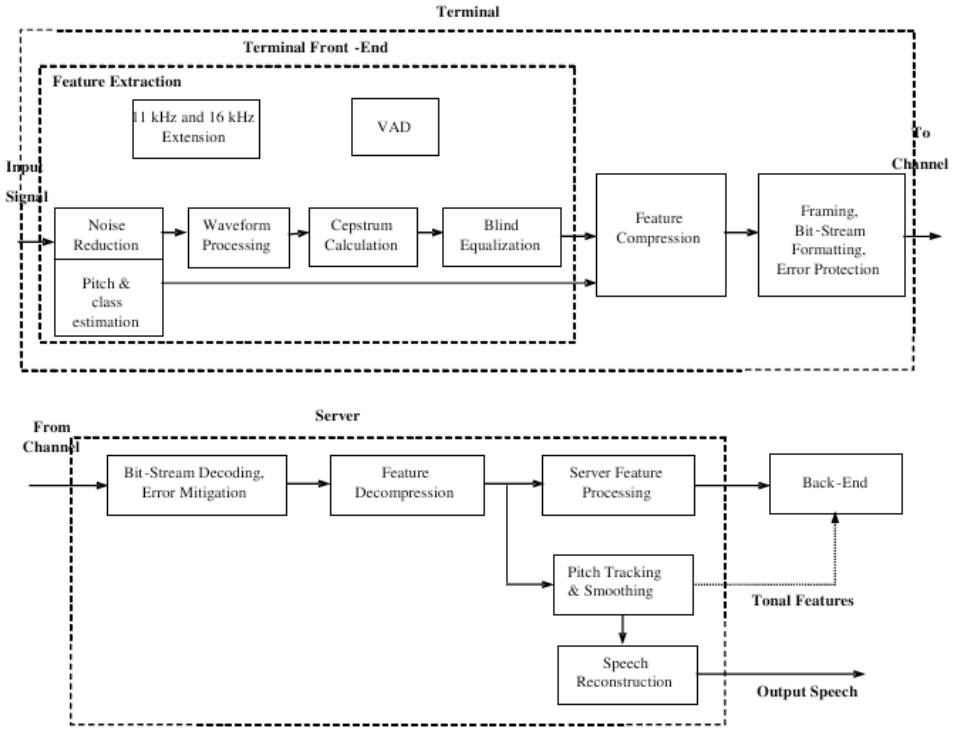

Figure 1. DSR scheme.

Noise reduction is based on Wiener filter theory and it is performed in two stages. The input signal is first de-noised in the first stage and the output of the first stage then enters the second stage. In the second stage, an additional, dynamic noise reduction is performed, which is dependent on the Signal-to-Noise Ratio (SNR) of the processed signal. Noise reduction is performed on a frame-by-frame basis. After framing the input signal, the linear spectrum of each frame is estimated in the Spectrum Estimation block. In PSD Mean block (Power Spectral Density), the signal spectrum is smoothed along the time (frame) index. Then, in the WF Design block, frequency domain Wiener filter coefficients are calculated by using both the current frame spectrum estimation and the noise spectrum estimation. The noise spectrum is estimated from noise frames, which are detected by a Voice Activity Detector (VADNest). Linear Wiener filter coefficients are further smoothed along

the frequency axis by using a Mel Filter-Bank, resulting in a Mel-warped frequency domain Wiener filter. The impulse response of this Mel-warped Wiener filter is obtained by applying a Mel IDCT (Mel-warped Inverse Discrete Cosine Transform). Finally, the input signal of each stage is filtered in the Apply Filter block. The input signal to the second stage is the output signal from the first stage. At the end of Noise Reduction, the DC offset of the noise-reduced signal is removed in the OFF block. SNR-dependent Waveform Processing (SWP) is applied to the noise reduced waveform that comes out from the Noise Reduction (NR) block. The noise reduction block outputs 80-sample frames that are stored in a 240-sample buffer (from sample 0 to sample 239). The waveform processing block is applied on the window that starts at sample 1 and ends at sample 200. Finally there is a block that performs cepstrum calculation. Cepstrum calculation is applied on the signal that comes out from the waveform processing block.

The second part of the research has focused on speaker adaptation, that will be discussed below.

3 Analisis and discussion of main results

Both MLLR and CMLLR use the EM criterion to estimate a linear transformation to adapt the Gaussian parameters of HMMs. Starting from the current set of parameters M, the adapted model parameters $\hat{M}$ are obtained by maximizing the following auxiliary function:

$$Q(M,\hat{M}) = K - \frac{1}{2}\sum_{m=1}^{M}\sum_{\tau=1}^{T}\left[K_m + \log\left|\hat{\Sigma}_m\right| + (o(\tau)-\hat{\mu}_m)^T \hat{\Sigma}_m^{-1}(o(\tau)-\hat{\mu}_m)\right] \quad (1)$$

where $\hat{\mu}_m$ and $\hat{\Sigma}_m$ are the adapted mean and variance of component m for the target acoustic condition while M and T represent respectively the number of components associated with the particular transform and the number of observations. K is a constant dependent only on the transition probabilities, K_m is the normalisation constant associated with Gaussian component m, and

$$\gamma_m(\tau) = p(q_m(\tau) \mid M, O_T) \quad (2)$$

is the posterior occupancy of component m, being $q_m(\tau)$ the Gaussian m at time τ and $O_T = [o(1),...,o(T)]$ the observation sequence.

3.1 Unconstrained transformation

In this adaptation method the mean and variance are transformed independently of each other. The mean μ is transformed as:

$$\hat{\mu} = A\mu + b = W\xi \quad (3)$$

where ξ is the extended mean vector, $\begin{bmatrix}1 & \mu^T\end{bmatrix}^T$, and $W = \begin{bmatrix}b & A\end{bmatrix}$ is the extended linear transform.

The transform of the covariance matrix Σ is given by:

$$\hat{\Sigma} = H\Sigma H^T \quad (4)$$

where H is the matrix to be estimated. Equation (1) represents the objective function to be maximized during adaptation to obtain the parameters W and H of the transformations. It was originally proposed to adapt the mean vector [4], extending the technique to

variance adaptation only later [7]. The mean based linear transform is referred to as MLLR, while covariance matrix transform is named variance MLLR.

3.2 Constrained transformation

The mean and the variance MLLR transformations can be simultaneously applied to both mean vectors and covariance matrices. However, as in this case the computational cost is high, a constrained scheme to adapt both mean vectors and covariance matrices can be used [4, 8]. This is referred to as constrained MLLR:

$$\hat{\mu} = A\mu + b, \tag{5}$$

$$\hat{\Sigma} = A\Sigma A^T \tag{6}$$

which is a particular case of unconstrained transformation with $H = A$. By substituting (5) and (6) into equation (1) and assuming a diagonal covariance matrix Σ, the following auxiliary function to be maximized is obtained:

$$Q(M, \hat{M}) = K - \frac{1}{2}\sum_{m=1}^{M}\sum_{\tau=1}^{T}\left[K_m + \log|\Sigma_m| - \log|A|^2 + (\hat{o}(\tau) - \mu_m)^T \Sigma_m^{-1}(\hat{o}(\tau) - \mu_m)\right] \tag{7}$$

where

$$\hat{o}(\tau) = A^{-1}o(\tau) - A^{-1}b = A_c o(\tau) + b_c = W_c \varsigma \tag{8}$$

as usual $W_c \equiv [b_c \quad A_c]$ represents the extended matrix transformation, and $\varsigma(\tau) = [1 \quad o(\tau)^T]^T$ is the extended vector of observations. Equation (7) clearly shows that the constrained transformation can be directly applied in the feature domain [9].

3.3 Iterative CMLLR

The proposed algorithm, referred to as iterative CMLLR (ICMLLR), is able to implement the constrained transformation using the standard auxiliary function (1) for MLLR, instead of maximizing the more complex objective function (7). The ICMLLR transform estimation is an iterative process: a first transformation W_0 is estimated by MLLR given an initial estimate of Σ, then at each iteration a new estimation $\hat{\Sigma}_k$ is forced to be

$$\hat{\Sigma}_k = A_k \hat{\Sigma}_{k-1} A_k^T \tag{9}$$

until convergence is reached. The algorithm proceeds as follows:

1. Assume an initial estimate Σ of $\hat{\Sigma}$.
2. Estimate the mean transformation $W_k = [b_k \quad A_k]$ by equation(1).
3. A new estimate is obtained by the constraint (9).
4. If a stop criterion on both $\hat{\mu}$ and $\hat{\Sigma}$ is not met, return to step 1.
5. Otherwise, if a stop criterion on both $\hat{\mu}$ and $\hat{\Sigma}$ is met, the solution is reached and the transformation (8) on feature domain can be applied.

3.4 Experimental results

In order to verify the effectiveness of the ICMLLR algorithm, experiments were conducted using the CMU Sphinx 4 ASR system, together with an advanced ETSI ES 202 212 feature extractor. The SI baseline model was generated according to the method described in [10]. All the experiments reported in this section were conducted using the first chapter of a long audiobook in Italian, whose audio and text transcriptions are freely available. In the first experiment, the iteration process was initialized by adapting the model on the first utterance of the test corpus, while the remaining utterances had been left available for recognition purposes. Each utterance was expected to be 40 phones long, corresponding to an average duration in time of about 500 ETSI frames (5.00 s). After initialization, the transform $W_c = [b_c \quad A_c]$ was estimated according the iteration process. Then the matrix W_c was used to transform the features as shown in equation (8). The estimation accuracy of ICMLLR was evaluated by performing several recognition tests, and comparing the results with those obtained by MLLR and conventional CMLLR. The behavior reported in Fig. 2 shows that the word error rate (WER), defined as the ratio of wrongly recognized or missing words to total words in the original text, reduces for both constrained algorithms as the number of iterations increases, slightly approaching the reference MLLR accuracy. It also must be noted that ICMLLR behaves always better than CMLLR.

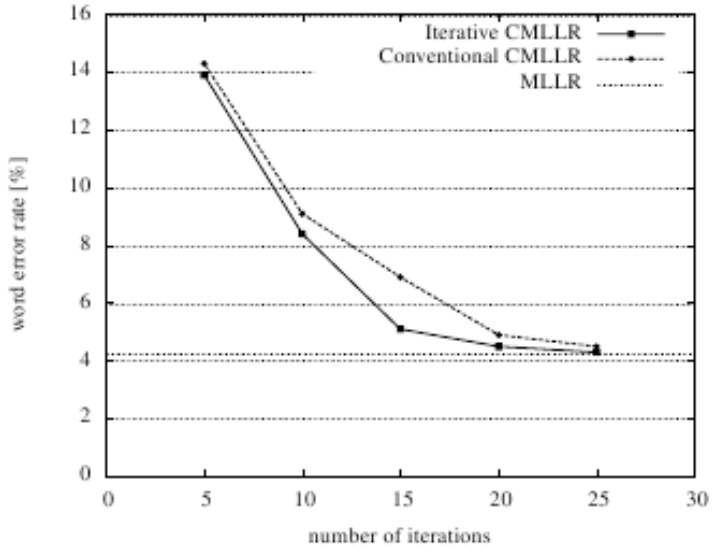

Figure 2. Accuracy evaluation as function of the number of iterations. Adaptation was carried out with the first test corpus utterance, while the remaining material was used for recognition purposes.

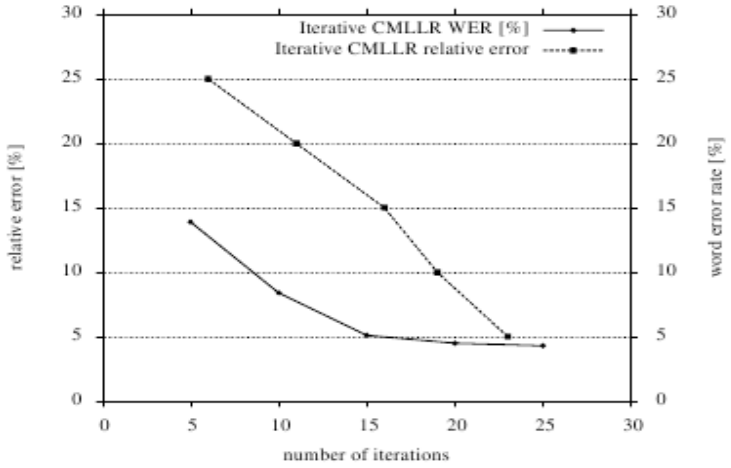

Figure 3. ICMLLR relative error and recognition accuracy evaluation as a function of the number of iterations. Test setup is the same as for Fig.2.

In addition, as a means to evaluate the ICMLLR convergence, the behavior of the relative error as a function of the number of iterations can be derived as well. The relative error for the iterative estimation of matrix W at step k is defined as

$$e_k\,\% = \frac{\| W_k - W_{k-1} \|}{\| W_1 - W_0 \|}$$

and can be used when defining a stopping criterion. Fig. 3 shows that the relative error decreases remarkably as the number of iterations increases, following the ICMLLR error rate trend. A second experiment was performed to assess whether the recognition accuracy improved using an increasing number of utterances as adaptation data. All the results were compared with those obtained with conventional CMLLR by setting the number of iterations for both algorithms at 15. As can be seen from Fig. 4, the ICMLLR approach shows better performance in terms of word error rate, and thus gives a better adaptation.

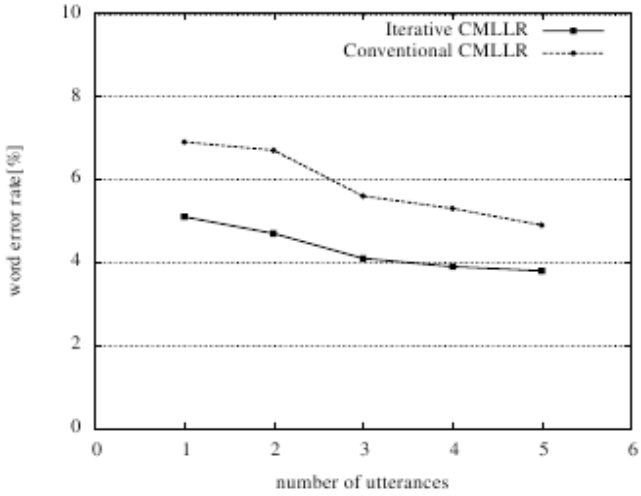

Figure 4. Accuracy comparison between ICMLLR and CMLLR. The number of iterations was fixed at 15 for both algorithms while letting adaptation data increase one utterance at time.

4 Conclusions

In this work, has been described the development of a distributed recognition system. It has been observed that the architecture of the recognition process in a distributed system is due to a scheme like client-server. For the purposes of speech recognition, the speech is not transmitted on a audio channel, but it is transmitted a coded parametric representation on a data channel. The front-end, placed in the client, calculates and extracts the parameters representative of speech (features) that are transmitted to the back-end placed on the server. Speech recognition is based on the processing of features stream.

ETSI (European Telecommunication Standard Institute) has proposed standard that allowed uniformity in implementations of speech recognition. The standard taken of reference for the development of the distributed speech recognition system is ETSI ES 202-212.

The front-end, capable of working in real time, was first implemented on the Java platform, and brought on a mobile device smartphone, based on Android platform. The remote back-end has been shaped by the Sphinx-4, an open source toolkit developed by Carnegie Mellon University.

We conducted an intensive experimentation from which have emerged results showed the effectiveness of the developed system. Based on the margins of improvement glimpsed, the thesis has subsequently focused on the study and implementation of speaker adaptation techniques. After careful analysis of the literature, in particular as regards the approaches of speaker adaptation, has been proposed and developed an original algorithm that implements the transformation bound for adaptation to the speaker, applicable client side. The adaptation techniques applicable to the talker client-side are suitable for use in distributed speech recognition systems because they have the advantage can be applied directly in the feature domain, allowing left intact, the acoustic model resident in the back-end, without violate any existing functionality in the core of the recognizer.

From the results emerged from the experimentation has been possible to establish that the algorithm developed is characterized by reduced complexity, while guaranteeing high performance. Moreover, the results obtained note the effectiveness of the proposed method and the functionality of the distributed speech recognition system as a whole.

References

[1] D. Povey and K. Yao. *A basis method for robust estimation of constrained MLLR*. Proc. 2011 IEEE International Conference on Acoustic, Speech and Signal Processing (ICASP-2011), Prague, Czech Republic, pp. 4460-4463, 2011.

[2] C. Legetter and P. Woodland. *Maximum likelihood linear regression for speaker adaptation of continuos density hidden Markov models*. Computer Speech and Language, vol. 9, No. 2, pp.171-185, 1995.

[3] M. Gales. *Maximum likelihood linear transformations for HMM-based speech recognition*. Comuter Speech and Language, vol. 12, No. 2, pp. 75-98, 1998.

[4] V. Digalakis, D. Rtischev and L. Neumeyer. *Speaker adaptation using constrained estimation of Gaussian mixtures*. IEEE Transaction on Speech and Audio Processing, Vol. 3, No. 5, pp. 357-366, 1995.

[5] L. Rabiner. *A tutorial on hidden Markov models and selected applications in speech recognition*. Proc. IEEE, vol. 77, No. 2 pp. 257-286, 1989.

[6] Y. Li, H. Erdogan, Y. Gao and E. Marcheret. *Incremental on-line feature space MLLR adaptation for telephony speech recognition.* Proc. 7th International Conference on Spoken Language Processing (ICSLP2002-Interspeech 2002), Denver, Colorado, pp. 1417-1420, 2002.

[7] M. Gales and P. Woodland. *Mean and variance adaptation within the MLLR framework.* Computer Speech and Language, vol. 10 , No. 4, pp. 249-264, 1996.

[8] M. Ferras, C.C. Leung, C. Barras and J. L. Gauvain. *Constrained MLLR for speaker recognition.* IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP 2007), Vol. 4, pp. 53-56, 2007.

[9] M. Ferras, C.C. Leung, C. Barras and J. L. Gauvain. *Comparison of speaker adaptation methods as feature extraction for SVM-based speaker recognition.* IEEE Transaction on Audio, Speech and Language Processing, Vol. 18, No. 6, pp. 1366-1378, 2010.

[10] M. Alessandrini, G. Biagetti, A. Curzi, C. Turchetti. *Semi-automatic acoustic model generation from large unsynchronized audio and text chuncks*, Proc. 12th Annual Conference of the International Speech Communication Association (Interspeech 2011), Florence, Itlay, pp. 1205-1208, 2011.