AN INEXPENSIVE PACKET LOSS COMPENSATION SCHEME FOR DISTRIBUTED SPEECH RECOGNITION BASED ON SOFT-FEATURES

Valentin Ion, Reinhold Haeb-Umbach

University of Paderborn
Dept. of Communications Engineering
33098 Paderborn, Germany
{ion,haeb}@nt.uni-paderborn.de

ABSTRACT

Soft-feature based speech recognition, which is an example of uncertainty decoding, has been proven to be a robust error mitigation method for distributed speech recognition over wireless channels exhibiting bit errors. In this paper we extend this concept to packet-oriented transmissions. The a posteriori probability density function of the lost feature vector, given the closest received neighbours, is computed. In the experiments, the nearest frame repetition, which is shown to be equivalent to the MAP estimate, outperforms the MMSE estimate for long bursts. Taking the variance into account at the speech recognition stage results in superior performance compared to classical schemes using point estimates. A computationally and memory efficient implementation of the proposed packet loss compensation scheme based on table lookup is presented.

1. INTRODUCTION

In a distributed speech recognition (DSR) scenario, the speech features computed at the client side, often a mobile device, are transmitted in some digital form over a communication channel to the remote speech recognition server.

A major research focus in recent years has been to overcome the degradation of speech recognition performance due to the unavoidable transmission errors. Several techniques, like interleaving and forward error correction at the client or splicing, repetition, interpolation at the server, have been proposed to mitigate the errors [1]. More elaborate algorithms [2], [3] attempt to utilize the channel reliability information and the inherent redundancy of the source to provide a so called soft-feature, consisting of a point estimate of the transmitted feature and the uncertainty about the estimation. The concealment of unreliable features occurs in the speech decoder by applying missing feature theory [4], weighted Viterbi decoding [5] or uncertainty decoding [6], the complement of Bayesian Predictive Classification (BCP) in the feature space [7].

In this paper we focus on obtaining the soft-features in a packet-switched communication scenario and effectively using them in the subsequent speech decoder. In the next section we present the general soft-feature framework for a continuous density HMM system. In Section 3 is shown how soft features can be computed for DSR over packet channels. A fast table lookup implementation is described in the Section 4. We show in Section 5 the experimental results obtained by our method on the Aurora 2 task and finish by drawing some conclusions in Section 6.

2. UNCERTAINTY DECODING

In a continuous density HMM system, the output distributions are represented by Gaussian mixture densities. For a given observation $\mathbf{x}_n$ at the time n, the output probability of the state s is expressed by:

$$p(\mathbf{x}_n|s) = \sum_{m=1}^{N_M} c_{sm}\mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_{sm}, \boldsymbol{\Sigma}_{sm}) \tag{1}$$

Here N_M is the number of Gaussians in the mixture and c_{sm} are the mixture weights.

In a DSR system, the feature vector $\mathbf{x}_n$ generated by the front-end is not known at the server due to transmission errors. Let $\boldsymbol{\zeta}$ denote the observations at the server side, which consist of a possibly corrupted version of $\mathbf{x}_n$ and future and past received feature vectors. The output probability $p(\mathbf{x}_n|s)$ of (1) now has to be replaced by:

$$\int_{\mathbf{x}} p(\mathbf{x}_n|s)p(\mathbf{x}_n|\zeta)d\mathbf{x}_n \tag{2}$$

i.e. the expected value of $p(\mathbf{x}_n|s)$, where the expectation is taken with respect to the a posteriori probability $p(\mathbf{x}_n|\zeta)$. This approach has been termed "uncertainty decoding" in [6].

However, the evaluation of the integral in the speech decoder for each state and observation vector is often beyond the available computing power. Some simplifying assumptions are necessary, although at the price of loosing optimality.

In our approach presented in [3], the posterior density was approximated by a Gaussian distribution $\mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_{\mathbf{x}_n|\boldsymbol{\zeta}}, \boldsymbol{\Sigma}_{\mathbf{x}_n|\boldsymbol{\zeta}})$, allowing to rewrite expression (2) as:

$$\sum_{m=1}^{M} c_{sm}\mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}_n|\boldsymbol{\zeta}}; \boldsymbol{\mu}_{sm}, \boldsymbol{\Sigma}_{sm} + \boldsymbol{\Sigma}_{\mathbf{x}_n|\boldsymbol{\zeta}}) \tag{3}$$

Comparing this with equation (1) we observe that the covariance of the state output probability is increased by $\boldsymbol{\Sigma}_{\mathbf{x}_n|\boldsymbol{\zeta}}$, the covariance of the posterior density, and that it is evaluated at $\boldsymbol{\mu}_{\mathbf{x}_n|\boldsymbol{\zeta}}$, the mean of the posterior density.

For an error free transmission $\boldsymbol{\Sigma}_{\mathbf{x}_n|\boldsymbol{\zeta}}$ is zero, denoting high reliability, and the observation probability does not change. For a completely corrupted transmission $\boldsymbol{\Sigma}_{\mathbf{x}_n|\boldsymbol{\zeta}}$ is very high and the posterior density becomes flat. The result of the integration (2) tends to be independent of HMM state in this case. Consequently, the contribution of this feature to the discrimination between acoustic models is reduced.

3. APPLICATION TO PACKET CHANNELS

3.1. Computation of posteriori probabilities

Let x_n be one component of the multidimensional feature vector $\mathbf{x}_n$ at the time n. In the front-end the value is quantized to the nearest centroid $c^{(i)}$, which is coded into a bit pattern $\mathbf{b}_n$ of length M. We write for clarity $\mathbf{b}_n = \mathbf{b}_n^{(i)}$ to denote that $\mathbf{b}_n$ represents the i-th centroid, $i = 1, \ldots, 2^M$. In the followings we consider that knowing $p(\mathbf{b}_n^{(i)}|\boldsymbol{\zeta})$ suffices to obtain the continuous-valued $p(x_n|\boldsymbol{\zeta})$.

The transmission of the bit pattern is modeled by an equivalent channel with input $\mathbf{b}_n^{(i)}$ and output $\hat{\mathbf{b}}_n$, characterized by the transmission probabilities $P(\hat{\mathbf{b}}_n|\mathbf{b}_n^{(i)})$. The prominent error pattern is considered to be packet loss. This means that either a packet is received, and then it is known to be error free:

$$P(\hat{\mathbf{b}}_n|\mathbf{b}_n^{(i)}) = P(\mathbf{b}_n^{(i)}|\hat{\mathbf{b}}_n) = \begin{cases} 1, & \text{if } \hat{\mathbf{b}}_n = \mathbf{b}_n^{(i)} \\ 0, & \text{if } \hat{\mathbf{b}}_n \neq \mathbf{b}_n^{(i)} \end{cases} \quad (4)$$

or it is lost, which means that:

$$P(\hat{\mathbf{b}}_n|\mathbf{b}_n^{(i)}) = \frac{1}{2^M} \quad (5)$$

We assume that the source is a Markov process, i.e. $P(\mathbf{b}_n|\mathbf{b}_{n-1}, \mathbf{b}_{n-2}, \ldots) = P(\mathbf{b}_n|\mathbf{b}_{n-1})$. Let us isolate a sequence of received bit patterns $\hat{\mathbf{B}} = (\hat{\mathbf{b}}_0, \ldots, \hat{\mathbf{b}}_{N+1})$ of which we know that the first $\mathbf{b}_0 = \mathbf{b}_0^{(s)}$ and last $\mathbf{b}_{N+1} = \mathbf{b}_{N+1}^{(e)}$ transmitted bit patterns have passed the channel uncorrupted. Considering $\zeta = \hat{\mathbf{B}}$, the computation of the a posteriori probability $P(\mathbf{b}_n^{(i)}|\hat{\mathbf{B}})$, $n = 1, \ldots, N$ can be accomplished by the forward-backward recursion [8]:

$$P(\mathbf{b}_n^{(i)}|\hat{\mathbf{B}}) = \frac{\alpha_n^{(i)}\beta_n^{(i)}}{\sum_{j=1}^{2^M}\alpha_n^{(j)}\beta_n^{(j)}} \quad (6)$$

where $i = 1, \ldots, 2^M$ and

$$\alpha_n^{(i)} = P(\mathbf{b}_n^{(i)}|\hat{\mathbf{b}}_0, \ldots, \hat{\mathbf{b}}_n) \quad (7)$$

$$\beta_n^{(i)} = P(\hat{\mathbf{b}}_{n+1}, \ldots, \hat{\mathbf{b}}_{N+1}|\mathbf{b}_n^{(i)}). \quad (8)$$

Both, $\alpha_n^{(i)}$ and $\beta_n^{(i)}$ can be computed recursively:

1. Initialization: $n := 0$

$$\alpha_0^{(i)} = P(\hat{\mathbf{b}}_0|\mathbf{b}_0^{(i)}) = \begin{cases} 1, & \text{if } i = s \\ 0, & \text{if } i \neq s \end{cases} \quad (9)$$

2. Recursion: for $n = 1 : N$

$$\alpha_n^{(i)} = \left[\sum_{j=1}^{2^M}\alpha_{n-1}^{(j)}P(\mathbf{b}_n^{(i)}|\mathbf{b}_{n-1}^{(j)})\right]P(\hat{\mathbf{b}}_n|\mathbf{b}_n^{(i)}) \quad (10)$$

Similarly, starting the recursion from the other end we obtain the backward probabilities.

1. Initialization: $n := N + 1$

$$\beta_{N+1}^{(i)} = 1 \quad (11)$$

2. Recursion: for $n = N : 1$

$$\beta_n^{(i)} = \sum_{j=1}^{2^M}P(\mathbf{b}_{n+1}^{(j)}|\mathbf{b}_n^{(i)})P(\hat{\mathbf{b}}_{n+1}|\mathbf{b}_{n+1}^{(j)})\beta_{n+1}^{(j)} \quad (12)$$

The a priori probabilities $P(\mathbf{b}_n^{(j)}|\mathbf{b}_{n-1}^{(i)})$ have been estimated in advance on a training speech database [3].

3.2. Matrix formulation of the forward-backward algorithm

In this section we give an efficient matrix formulation of the forward-backward algorithm. First define the row vector of size 2^M:

$$\boldsymbol{\alpha}_0 = P(\hat{\mathbf{b}}_0|\mathbf{b}_0^{(i)}) = (0, 0, \ldots, 0, 1, 0, \ldots, 0) \quad (13)$$

which consist of zeros except for $P(\hat{\mathbf{b}}_0|\mathbf{b}_0^{(s)}) = 1$ at s-th position.

Similarly define the row vector:

$$\boldsymbol{\beta_{N+1}} = (0, 0, \ldots, 0, 1, 0, \ldots, 0) \quad (14)$$

of the same size as $\boldsymbol{\alpha}_0$, where 1 is now at the e-th position. Further define the $(2^M \times 2^M)$-dimensional matrix of a priori probabilities $\mathbf{A}$, where the element on the i-th row and j-th column is $(\mathbf{A})_{ij} = P(\mathbf{b}_n^{(j)}|\mathbf{b}_{n-1}^{(i)})$.

Noting that (5) holds for the duration of the burst, $n = 1, \ldots, N$, and that (4) holds for $n = 0$ and $n = N + 1$, (10) and (12) can be simplified, and the forward-backward probabilities can now be computed as follows ($n = 1, \ldots, N$):

$$\boldsymbol{\alpha}_n = \boldsymbol{\alpha}_{n-1} \cdot \mathbf{A} = \boldsymbol{\alpha}_0 \cdot \mathbf{A}^n \quad (15)$$

$$\boldsymbol{\beta}_n = \boldsymbol{\beta}_{n+1} \cdot \mathbf{A}^T = \boldsymbol{\beta}_{N+1} \cdot (\mathbf{A}^{N-n+1})^T \quad (16)$$

Since $\mathbf{A}$ consists of a priori probabilities, $\mathbf{A}^n$ can be computed in advance and stored, thus saving a lot of computations during runtime. Although this seems to come at the price of increased memory demands, we will show in the following how this can be avoided.

3.3. Mutual information

A closer look at equation (15) reveals that $(\mathbf{A}^k)_{ij} = P(\mathbf{b}_n^{(j)}|\mathbf{b}_{n-k}^{(i)})$. Table 1 gives the mutual information $I(\mathbf{b}_n; \mathbf{b}_{n-k}) = H(\mathbf{b}_n) - H(\mathbf{b}_n|\mathbf{b}_{n-k})$, where $H(\mathbf{b}_n)$ denotes the entropy of the bit pattern $\mathbf{b}_n$. $I(\mathbf{b}_n; \mathbf{b}_{n-k})$ is a measure of the information about $\mathbf{b}_n$, that is contained in $\mathbf{b}_{n-k}$ and thus indicates whether it is useful to utilize $\mathbf{b}_{n-k}$ for the reconstruction of $\mathbf{b}_n$. The values have been obtained using the ETSI front-end on the Aurora 2 training set. Note that ETSI standard uses split vector quantization and each subvector ($sv_{1,\ldots,7}$) is coded separately into a bit pattern. It can be seen that $I(\mathbf{b}_n; \mathbf{b}_{n-k})$ tends to zero as k increases. This means that there is a depth L for which $H(\mathbf{b}_n|\mathbf{b}_{n-k}) \simeq H(\mathbf{b}_n)$ for $k \geq L$. Note that if the conditional entropy equals the unconditional, the rows of A^k become constant [9, p.161].

Table 1. *Entropies and mutual information among the b_n and b_{n-k} produced by the ETSI advanced DSR front-end.*

Subvector	sv_1	sv_2	sv_3	sv_4	sv_5	sv_6	sv_7
M	6	6	6	6	6	5	8
$H(b_n)$	5.8	5.8	5.8	5.8	5.8	4.8	7.7
$I(b_n; b_{n-1})$	2.6	2.1	1.6	1.4	1.2	1.0	3.4
$I(b_n; b_{n-2})$	1.7	1.3	0.9	0.8	0.7	0.6	2.8
$I(b_n; b_{n-3})$	1.2	0.9	0.7	0.6	0.5	0.4	2.1
$I(b_n; b_{n-4})$	0.9	0.7	0.5	0.4	0.3	0.3	1.8
$I(b_n; b_{n-5})$	0.7	0.5	0.3	0.3	0.2	0.2	1.4

3.4. Frame reconstruction

The reconstruction of x_n reduces to finding the proper parameters $\mu_{x_n|\boldsymbol{\zeta}}$ and $\sigma^2_{x_n|\boldsymbol{\zeta}}$ of a Gaussian density that approximates the

discrete distribution $p(x_n^{(i)}|\zeta), i = 1 \dots 2^M$, where $x_n^{(i)} = c^{(i)}$ is the i-th codebook centroid coresponding to the bit pattern $\mathbf{b}_n^{(i)}$. We experimentally observed that in the first half of the burst, i.e. $n = 1, \dots, \frac{N}{2}$, the maximum of $p(x_n^{(i)}|\zeta)$ is in the most cases at $i = s$ which means that the most probable value of x_n given ζ is the last correctly received before the burst. Similarly, $x_n^{(e)} = c^{(e)}$ is the most probable value in the second half of the burst. This observation indicates that nearest frame repetition (NFR) is in fact a MAP estimation strategy. Therefore an option for estimating x_n is NFR.

Another option we tried was the MMSE estimate:

$$\mu_{x_n|\boldsymbol{\zeta}} = \sum_{i=1}^{2^M} c^{(i)} \cdot p(x_n^{(i)}|\zeta). \tag{17}$$

The variance of the distribution is obtained by:

$$\sigma^2_{x_n|\boldsymbol{\zeta}} = \sum_{i=1}^{2^M} (c^{(i)} - \mu_{x_n|\boldsymbol{\zeta}})^2 \cdot p(x_n^{(i)}|\zeta). \tag{18}$$

As an approximation, the MAP estimate can be used in (18) in place of $\mu_{x_n|\boldsymbol{\zeta}}$.

4. FAST TABLE LOOKUP IMPLEMENTATION

The drawback of the computationally efficient algorithm presented in Section 3 is that the matrices $\mathbf{A}^n, n = 1, \dots, L$ have to be stored. For the quantization scheme used in ETSI Front-end for DSR [10] this amounts to $L \times (5 \cdot 2^{2\cdot 6} + 2^{2\cdot 5} + 2^{2\cdot 8}) = L \times 87040$ values. We used $L = 6$ in our experiments. In this section we propose a simplification which results in significant memory and computational savings without loss of performance. Let

$$\mathbf{c} = [c^{(1)}, c^{(2)}, \dots, c^{(2^M)}] \tag{19}$$

be the vector of codebook centroids and

$$\mathbf{c^2} = [(c^{(1)})^2, (c^{(2)})^2, \dots, (c^{(2^M)})^2] \tag{20}$$

the vector of their squared values. Ignoring the contribution of the backward recursion, we use the NFR estimate $x_n^{(s)}$, $n = 1, \dots, \frac{N}{2}$ in the first half of the burst and compute the variance of the posterior probability by:

$$\begin{aligned} \sigma_n^2 &= \sum_{i=1}^{2^M} (c^{(i)} - x_n^{(s)})^2 \cdot \alpha_n^{(i)} \\ &= \mathbf{c^2} \cdot (\mathbf{A}^n)^T \cdot \boldsymbol{\alpha}_0^T - 2x_n^{(s)} \mathbf{c} \cdot (\mathbf{A}^n)^T \cdot \boldsymbol{\alpha}_0^T + (x_n^{(s)})^2 \end{aligned} \tag{21}$$

For the second half of the burst we use $x_n^{(e)}$ as estimate for the mean and simply use the variance computed on the first half. The advantage is that the expressions $\mathbf{c} \cdot (\mathbf{A}^n)$ and $\mathbf{c^2} \cdot (\mathbf{A}^n)$ are vectors of length 2^M, which need considerably less storage than $\mathbf{A}^n$ and which can be computed prior to recognition. Moreover, because $\boldsymbol{\alpha}_0$ is a vector of zeros except of a one at position s, the multiplication of a vector with $\boldsymbol{\alpha}_0^T$ results in simply selecting the s-th element of that vector. For the ETSI quantization scheme the memory requirement is reduced from $L \times 87040$ to $L \times 1216$.

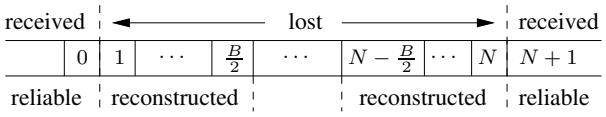

Fig. 1. *Schematic drawing of the burst of lost packets.*

5. EXPERIMENTAL RESULTS

This section presents the results of the test we performed in order to evaluate the effectiveness of the approach and to see the influence of the approximations we have made on the recognition word error rate. We have simulated a packet oriented transmission for various network conditions. Each packet consisted of two feature vectors. The losses have been induced by using a 2-states Markov chain [11], characterized by the conditional loss probability clp and mean loss probability mlp.

Table 2. *The conditional loss probability and mean loss probability of the four simulated network conditions.*

Condition	C1	C2	C3	C4
clp	0.147	0.33	0.5	0.6
mlp	0.006	0.09	0.286	0.385

The recognition task was the clean set of AURORA 2 database consisting of 4004 utterances distributed over 4 subsets and the acoustic models were those described in [12].

The ETSI advanced front-end for DSR [10] was employed for feature extraction and quantization. The word error rate in the error free scenario was 0.86% for this setup.

5.1. Reconstruction without uncertainty decoding

We evaluate first the so called "plug-in" methods, where the lost value is replaced by a point estimate which is fed into the unmodified speech recognizer as if it were the true sent value. The left side of Table 3 shows the word error rates (WER) when using NFR for reconstruction. This, according to the notations of 3.1, means repetition of $\hat{\mathbf{b}}_0$ in the first half of the burst and $\hat{\mathbf{b}}_{N+1}$ in the second. The right side of the table shows the WER achieved by MMSE reconstruction.

The value B in the Tables 3 and 4 denotes the maximal number of frames in a burst for which the reconstruction is carried out. That means, for bursts longer than B only the first and last $\frac{B}{2}$ frames are reconstructed (see Figure 1). The reason to do so is showing that MMSE reconstruction can do better that NFR for short bursts. Looking only at the last line ($B = 48$) of the table, we would conclude that NFR outperforms MMSE. However, the frames close to one end of the burst are much better reconstructed by MMSE, as the line $B = 6$ shows. These results support what we also noted in Section 3.3: the farther the reconstructed frame is from one end of the burst, the smaller the mutual information becomes, making the correct reconstruction more difficult.

5.2. Uncertainty decoding

In the second set of experiments we utilized a modified speech decoder taking into account the unreliability of a feature as presented

Table 3. *Word error rates* [%] *for NFR and MMSE reconstruction as a function of maximal number of reconstructed frames* B *for different channel conditions*

	NFR				MMSE			
B	C1	C2	C3	C4	C1	C2	C3	C4
6	0.86	1.12	2.75	6.45	0.86	1.11	2.35	5.33
12	0.86	1.06	2.33	5.02	0.86	1.08	2.26	4.79
24	0.86	1.06	2.30	5.01	0.86	1.08	2.32	5.44
48	0.86	1.06	2.31	5.02	0.86	1.08	2.32	5.47

in Section 2. The same reconstruction methods NFR and MMSE have been evaluated again. The WERs are tabulated in Table 4.

Table 4. *Word error rates* [%] *for NFR and MMSE reconstruction with uncertainty decoding as a function of maximal number of reconstructed frames* B *for different channel conditions*

	NFR				MMSE			
B	C1	C2	C3	C4	C1	C2	C3	C4
6	0.86	1.03	2.13	5.01	0.86	1.03	2.14	4.74
12	0.86	1.01	1.82	3.78	0.86	1.01	2.04	4.45
24	0.86	1.01	1.80	3.72	0.86	1.01	2.07	4.66
48	0.86	1.01	1.80	3.72	0.86	1.01	2.08	4.66

Overall, uncertainty decoding is superior to the plug-in methods. However, the same trend is observed: when used with MMSE, it is effective only for the reconstruction of the frames close to one end of the burst while the far-away frames are reconstructed poorly.

While the results presented sofar had been obtained by the exact forward-backward algorithm, Table 5 gives the results using the fast table lookup method of Section 4. Surprisingly, no performance loss occurred compared to the corresponding results in Table 4, on the contrary it seems to offer a slight improvement. However, we cannot explain this behavior at this time.

Table 5. *Word error rates* [%] *for NFR reconstruction and uncertainty decoding with reliability computed by table lookup.*

B	C1	C2	C3	C4
6	0.86	1.06	2.16	5.10
12	0.86	1.03	1.80	3.72
24	0.86	1.03	1.82	3.59
48	0.86	1.03	1.82	3.60

6. CONCLUSIONS

In this work we developed and tested a soft-feature concept for distributed speech recognition over loosy packet channels. It is observed that the MMSE estimate of the lost frame works best for short error bursts while nearest frame repetition, which is shown to be closely related to the MAP estimate, is superior for longer bursts. In both cases the soft information which is utilized in uncertainty decoding, gives significant performance improvement, e.g. a 30% reduction of WER on a channel with 40% packet loss ratio (condition C4). Using constraints which are specific to a packet loss scenario, we demonstrated how the soft features can be easily obtained using a moderately sized lookup table and involving virtually no computational effort.

7. ACKNOWLEDGEMENTS

This work was supported by Deutsche Forschungsgemeinschaft under contract number HA 3455/2-1.

8. REFERENCES

[1] Z.-H. Tan, P. Dalsgaard, and B. Lindberg, "Automatic speech recognition over error-prone wireless networks," *Speech Communication*, vol. 47, no. 1-2, pp. 220–242, Sep.-Oct. 2005.

[2] V. Weerackody, W. Reichl, and A. Potamianos, "An error-protected speech recognition system for wireless communications," *IEEE Trans. Wireless Communications*, vol. 1, no. 2, pp. 282–291, 2002.

[3] R. Haeb-Umbach and V. Ion, "Soft features for improved distributed speech recognition over wireless networks," in *Proc. of ICSLP, Jeju, Korea*, 2004.

[4] T. Endo, S. Kuroiwa, and S. Nakamura, "Missing feature theory applied to robust speech recognition over IP networks," in *Proc. of EUROSPEECH, Geneva, Switzerland*, 2003.

[5] A. Bernard and A. Alwan, "Low-bitrate distributed speech recognition for packet-based and wireless communication," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 8, pp. 570–579, 2002.

[6] L. Deng, J. Droppo, and A. Acero, "Dynamic compensation of HMM variances using the feature enhancement uncertainty computed from a parametric model of speech distortion," *IEEE Trans. Speech and Audio Processing*, vol. 13, no. 3, pp. 412–421, 2005.

[7] Q. Huo and C-H. Lee, "A Bayesian predictive approach to robust speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 8, pp. 200–204, 2000.

[8] A.M. Peinado, V. Sanchez, J.L. Perez-Cordoba, and A. de la Torre, "HMM-based channel error mitigation and its application to distributed speech recognition," *Speech Communication*, vol. 41, no. 6, pp. 549–561, Nov. 2003.

[9] Mitrani I., *Probabilistic modelling*, Cambridge University Press, 1998.

[10] ETSI, "ES 202 050 v1.1.1., Speech processing, Transmission and Quality aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms," *Tech. rep. ETSI*, Oct 2002.

[11] C. Boulis, M. Ostendorf, E:A. Riskin, and S. Otterson, "Gracefully degradation of speech recognition performance over packet-erasure networks," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 8, pp. 580–590, Nov. 2002.

[12] H. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ISCA ITRW Workshop ASR2000, Paris, France*, 2000.