Improved Source Modeling and Predictive Classification for Channel Robust Speech Recognition

Valentin Ion, Reinhold Haeb-Umbach

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

Abstract

The accuracy of distributed speech recognition has been shown to be very sensitive to errors occurring during transmission. One reason for this is that the classifier, usually trained under error free conditions, is unable to cope with the mismatch between an error free and error prone channel. In this paper we present a novel decision rule for classification which is able to account for channel errors. To achieve this, the classical Bayesian speech recognition approach has been reformulated for the server side, where the observation is known only to the extent, as is given by its a posteriori density function. We present a method to estimate the a posteriori density which is based on a Markov model of the source, which captures correlations of both static and dynamic features. A practical implementation is given, accompanied by experimental results for distributed speech recognition over an IP-network.

Index Terms: distributed speech recognition, channel robustness, predictive classification.

1. Introduction

The client-server architecture of distributed speech recognition (DSR) enables a mobile device to access sophisticated speech recognition services without the need to run and maintain complex speech recognition software and application data. A front-end, running on the mobile client, extracts the speech features, compresses them, protects them against channel errors and sends them over a communication link to the backend server, where decoding, uncompression and automatic speech recognition (ASR) takes place.

Since the recognition accuracy is severely affected by transmission errors, many proposals have been made to improve over the error mitigation scheme originally proposed in the ETSI standard ES 202 050, see [1] and the references therein. Some of them may be categorized as point estimation techniques. They attempt to correct or reconstruct the erroneous data using either forward error correction or the redundancy of the data. Others modify the ASR-decoder in order to deemphasize the contribution of those feature vectors on the classification, which are deemed erroneous.

In our prior work we have cast the channel error mitigation problem in a probabilistic formulation. We have developed a practical approach to compute so-called soft-features, i.e. the a posteriori probability of the original error-free feature vector, given all received feature vectors, and used them in the uncertainty decoding rule, which has been proposed by Deng et al. for noise-robust speech recognition [2]. This resulted in improved channel error
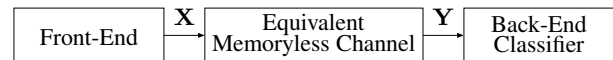

Figure 1: *Block diagram of a distributed recognition system.*

robustness compared to the error mitigation proposed in the ETSI standard, both for circuit-switched and packet-switched transmission.

In this paper we go one step further and derive a predictive classification rule, given the feature vectors at the server side, and show by experimental evidence that this novel rule outperforms the uncertainty decoding rule. Further, the Markov model of the error-free feature vector sequence is extended in order to capture the correlation between both consecutive static and dynamic features, which results in additional performance improvements.

In the next section we present the novel predictive decision rule, and Section 3 details practical aspects on how to use this classifier in the context of DSR over a channel characterized by packet losses. Section 4 presents the experimental results and the paper finishes with conclusions drawn in Section 5.

2. Predictive Decision Rule

2.1. Bayesian Speech Recognition

We summarize very shortly the classification problem for recognizing continuous speech: Given the sequence of feature vectors $\mathbf{X}$ extracted from an utterance, the statistical speech recognition attempts to find the sequence of words $\hat{\mathbf{W}}$ out of a given vocabulary which maximizes the probability $P(\mathbf{W}|\mathbf{X})$. Using the Bayesian theorem this can be expressed more conveniently as maximizing the product between observation probability and word sequence probability:

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\operatorname{argmax}}\; p(\mathbf{X}|\mathbf{W})P(\mathbf{W}), \qquad (1)$$

where both probability terms can be estimated in a training phase, prior to recognition.

2.2. Predictive Classification

We consider the distributed speech recognition system depicted in Figure 1.

The feature extraction at the client side delivers the sequence of feature vectors $\mathbf{X}$, whereas at the classifier input the received sequence $\mathbf{Y}$ is available. Instead of plugging-in the received vectors into the recognizer as if they were the sent ones, we reformulate
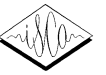
the classification to be performed with the received vectors. Therefore, the problem reduces now to finding the word sequence which satisfies the relation:

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\operatorname{argmax}} \; p(\mathbf{Y}|\mathbf{W})P(\mathbf{W}). \quad (2)$$

Here, the probability of observing $\mathbf{Y}$ given $\mathbf{W}$ cannot be obtained by training any more, as it certainly depends on the, possibly time-varying, channel properties, not known at training time.

The way around this problem that we take here is to express $P(\mathbf{W}|\mathbf{Y})$ as a predictive density function [3]:

$$P(\mathbf{W}|\mathbf{Y}) = \int P(\mathbf{W}|\mathbf{X}) \cdot p(\mathbf{X}|\mathbf{Y})d\mathbf{X} \quad (3)$$

Here, we have used the fact that $P(\mathbf{W}|\mathbf{X}, \mathbf{Y}) = P(\mathbf{W}|\mathbf{X})$, since the received vectors do not contain more information about the words than is already present in the sent vectors.

Thus, applying now the Bayes theorem we obtain the predictive decision rule:

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\operatorname{argmax}} \; \int p(\mathbf{X}|\mathbf{W})\frac{p(\mathbf{X}|\mathbf{Y})}{p(\mathbf{X})}d\mathbf{X} \cdot P(\mathbf{W}) \quad (4)$$

Note that an expression similar to (4) was also used in other works [2], [9], however the a priori $p(\mathbf{X})$ had been neglected. We show that in our case this term is very important and significant improvement is obtained when considering it.

By modeling the words as sequences of states $\mathbf{S} = (s_1, \ldots, s_T)$ in a Hidden Markov Model (HMM) and using the so-called maximum approximation, the observation probability $p(\mathbf{X}|\mathbf{W})$, where $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$, can be computed by the Viterbi algorithm. In the case of decision rule (1) one needs to compute

$$p(\mathbf{X}|\mathbf{W}) \approx \underset{\mathbf{S}}{\max} \prod_{t=1}^{T} P(s_t|s_{t-1})p(\mathbf{x}_t|s_t). \quad (5)$$

Using the new decision rule (4) one obtains:

$$\int p(\mathbf{X}|\mathbf{W})\frac{p(\mathbf{X}|\mathbf{Y})}{p(\mathbf{X})}d\mathbf{X} \approx \underset{\mathbf{S}}{\max} \prod_{t=1}^{T} P(s_t|s_{t-1})q(\mathbf{Y}; s_t) \quad (6)$$

where

$$q(\mathbf{Y}; s_t) = \int p(\mathbf{x}_t|s_t)\frac{p(\mathbf{x}_t|\mathbf{Y})}{p(\mathbf{x}_t)}d\mathbf{x}_t \quad (7)$$

This decision rule requires knowledge of the posteriori density $p(\mathbf{x}_t|\mathbf{Y})$. In the next section we present an approach how to determine this term and how to evaluate (7) without expensive numerical integration.

3. Practical aspects

3.1. Computation of the a posteriori density function

Let $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_T)$ be the sequence of received feature vectors, where the lost frames in case of packet-oriented transmission are recreated at random so that the sequences $\mathbf{X}$ and $\mathbf{Y}$ have the same length. Each feature vector at time t consists of static and dynamic (velocity and acceleration) components: $\mathbf{x}_t = (\tilde{\mathbf{x}}_t, \tilde{\mathbf{x}}'_t, \tilde{\mathbf{x}}''_t)$. Note that only the static components are transmitted, while the dynamic components are computed from received static components.

Our goal is to compute the conditional probability $p(\mathbf{x}_t|\mathbf{Y})$, i.e. the a posteriori probability density function of the sent vector at each time t when the received sequence $\mathbf{Y}$ is known. In the following, we employ models of increasing complexity to determine this posterior.

3.1.1. Memoryless source

A very simplistic approach is to consider the signal source be memoryless. It can be easily shown that this is equivalent to $p(\mathbf{x}_t|\mathbf{Y}) = p(\mathbf{x}_t|\mathbf{y}_t)$, if we assume the channel to be memoryless. The probability term (7) then simplifies to:

$$\int p(\mathbf{x}_t|s_t)\frac{p(\mathbf{x}_t|\mathbf{y}_t)}{p(\mathbf{x}_t)}d\mathbf{x}_t \quad = \quad \int p(\mathbf{x}_t|s_t)\frac{p(\mathbf{y}_t|\mathbf{x}_t)}{p(\mathbf{y}_t)}d\mathbf{x}_t \quad (8)$$

If the channel is error free, $p(\mathbf{y}_t|\mathbf{x}_t) = \delta(\mathbf{x}_t - \mathbf{y}_t)$ and (8) reduces to the ordinary observation probability $p(\mathbf{x}_t|s_t)$, since the constant $p(\mathbf{y}_t)$ is irrelevant. During packet loss, on the other hand, $p(\mathbf{y}_t|\mathbf{x}_t) = p(\mathbf{y}_t)$ which reduces (8) to unity, resulting in a marginalization of the features [4].

3.1.2. Static components modeled as Markov source

The approximation made in the previous section has the disadvantage that it ignores the correlation between consecutive vectors. It is expected that a lost vector can be retrieved to some extent if the predecessor vector is known. To check that, we analyzed the entropies and mutual information of the cepstral features extracted using the advanced front-end for distributed speech recognition standardized by ETSI [5]. The static component $\tilde{\mathbf{x}}_t$ consists of seven subvectors, each one grouping two cepstral coefficients. Table 1 lists the entropies and mutual information for each subvector $\tilde{\mathbf{v}}$.

Table 1: *Entropies and mutual information among the subvectors produced by the ETSI advanced DSR front-end.*

Subvector	1	2	3	4	5	6	7
M	6	6	6	6	6	5	8
$H(\tilde{\mathbf{v}}_\mathbf{t})$	5.8	5.8	5.8	5.8	5.8	4.8	7.7
$I(\tilde{\mathbf{v}}_\mathbf{t}; \tilde{\mathbf{v}}_{\mathbf{t-1}})$	2.6	2.1	1.6	1.4	1.2	1.0	3.4
$I(\tilde{\mathbf{v}}_\mathbf{t}; \mathbf{v}_{\mathbf{t-1}})$	3.0	2.4	1.9	1.7	1.5	1.3	4.5

Each subvector was quantized individually by a split vector quantization scheme resulting in seven bit patterns of length M. The mutual information $I(\tilde{\mathbf{v}}_t; \tilde{\mathbf{v}}_{t-1})$ indicates how much information about the current subvector $\tilde{\mathbf{v}}_\mathbf{t}$ is already present in the previous subvector $\tilde{\mathbf{v}}_{t-1}$.

It is therefore reasonable to consider $\tilde{\mathbf{x}}_t$ a Markov process which can be described by Hidden Markov Model (HMM) theory [6, p.321]. From now on, we denote by $\tilde{\mathbf{x}}_t$ any of the subvectors. Let $\tilde{\mathbf{x}}_t \in Q$ be the quantized subvector, where $Q = \{\tilde{\mathbf{x}}^{(\mathbf{i})}|i = 1, \ldots, 2^M\}$, is the set of $N = 2^M$ codebook centroids. A complete specification of a HMM is given by the set of states, the set of observation symbols and three sets of probability measures A, B, π, where A is state-transition probability distribution, B is the observation symbol probability distribution and π denotes the initial state distribution. We consider the following HMM:

- $N = 2^M$ states $\tilde{\mathbf{x}}^{(i)}; i = 1, \ldots, N$
- N output symbols per state $\tilde{\mathbf{y}}_t \in Q$
- $A = a_{ij} = P(\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}^{(j)}|\tilde{\mathbf{x}}_{t-1} = \tilde{\mathbf{x}}^{(i)}); i, j = 1, \ldots, N$
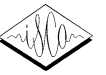
- $B = b_j(k) = P(\tilde{\mathbf{y}}_t = \tilde{\mathbf{x}}^{(k)} | \tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}^{(j)}); k, j = 1, \ldots, N$

- π is an uniform distribution

Note that we use uppercase $P(\cdot)$ to denote the probability mass function of a discrete random variable and lowercase $p(\cdot)$ to denote the coresponding density consisting of a finite sum of Dirac pulses. The term $P(\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}^{(i)} | \tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_T)$ can now be computed by the forward-backward algorithm. Note that this procedure was used in [7] in order to get the most probable "state" sequence, whereas we are here interested in the posterior state distribution rather than a point estimate.

3.1.3. Improved source modeling

The assumption of a Markov source for $\tilde{\mathbf{x}}_t$ is certainly far from being true. Experiments have shown that the dependency cannot be reduced only to the previous frame. Actually, $\tilde{\mathbf{x}}_t$ depends also on $\tilde{\mathbf{x}}_{t-2}, \tilde{\mathbf{x}}_{t-3}$ etc. A trade off between complexity and accuracy is to consider the whole vector, i.e. static and dynamic components, $\mathbf{x}_t = (\tilde{\mathbf{x}}_t, \tilde{\mathbf{x}}_t', \tilde{\mathbf{x}}_t'')$ to be generated by a Markov source. In fact, the dynamic components hide some longer time span dependencies. The last line of Table 1 lists the mutual information among $\tilde{\mathbf{v}}_t$ and $\mathbf{v}_{t-1}$, which indicates how much is known about the static component of the current frame when the static and dynamic components of the previous frame are known. In order to compute this, the dynamic components were also vector quantized using $D_1 = 3$ bits for velocity and $D_2 = 1$ bit for acceleration.

It can be observed that more redundancy can be exploited if the state space of the HMM defined above is extended to represent both static and dynamic feature vector components. We therefore build following HMM with an extended state space:

- $N = 2^{M+D_1+D_2}$ states $\mathbf{x}^{(i)}; i = 1, \ldots, N$

- 2^M output symbols per state $\tilde{\mathbf{y}}_\mathbf{t} \in Q$. Note that only the static components are observed, because the delta and acceleration are not transmitted.

- $A = a_{ij} = P(\mathbf{x}_t = \mathbf{x}^{(j)} | \mathbf{x}_{t-1} = \mathbf{x}^{(i)}); i, j = 1, \ldots, N$

- $B = b_j(k) = P(\tilde{\mathbf{y}}_t = \tilde{\mathbf{x}}^{(k)} | \mathbf{x}_t = \mathbf{x}^{(j)}); k = 1, \ldots, 2^M, j = 1, \ldots, N$

- π is an uniform distribution

Because $\tilde{\mathbf{y}}_t$ corresponds to the received static components $\tilde{\mathbf{x}}_t$, it is expected to depend only on these, so that:

$$p(\tilde{\mathbf{y}}_t | \mathbf{x}_t) = p(\tilde{\mathbf{y}}_t | \tilde{\mathbf{x}}_t) \qquad (9)$$

The right side of (9) is either a Dirac pulse, when the channel is error free, or an uniform distribution, when packets are lost. In a circuit switched environment, it might also have another functional form, depending on the individual bit error probabilities [8].

Now the forward-backward recursion can again be used to compute the posterior $p(\mathbf{x}_t = \mathbf{x}^{(i)} | \mathbf{Y})$, $i = 1, \ldots, N$. From this, the posterior of static components can be obtained by:

$$p(\tilde{\mathbf{x}}_t | \mathbf{Y}) = \int\int p(\mathbf{x}_t | \mathbf{Y}) d\tilde{\mathbf{x}}_t' d\tilde{\mathbf{x}}_t''. \qquad (10)$$

The means of the posterior densities of velocity and acceleration are computed from the means of $p(\tilde{\mathbf{x}}_t | \mathbf{Y})$, $t = 1, \ldots, T$ and the variances are computed assuming that $\ldots, \tilde{\mathbf{x}}_{t-1} | \mathbf{Y}, \tilde{\mathbf{x}}_t | \mathbf{Y}, \tilde{\mathbf{x}}_{t+1} | \mathbf{Y}, \ldots$ are independent random variables, certainly a simplifying assumption [9].

We also tried alternatively to obtain the distribution of dynamic components directly from the state distribution, similar to (10), but the results were not encouraging, probably due to the rough quantization for delta and acceleration that we employed.

3.2. Integration in the recognition engine

As already explained in section 3.1.1, in the extreme cases of an error-free channel and a channel allowing no information transmission at all, the posterior $p(\mathbf{x}_t | \mathbf{Y})$ reduces to a Dirac pulse centered at the received $\mathbf{y}_t$ or to the a priori density $p(\mathbf{x}_t)$, respectively. For both extreme cases, as well for all intermediate channel conditions we approximate the a posteriori density by a Gaussian density with the same time varying mean and variance as the original discrete posterior density.

While the validity of this assumption is certainly debatable, it allows an analytic computation of the integral (7) using the following formula:

$$\int \mathcal{N}(x; \mu_1, \sigma_1^2) \cdot \frac{\mathcal{N}(x; \mu_2, \sigma_2^2)}{\mathcal{N}(x; \mu_3, \sigma_3^2)} dx = C \cdot \mathcal{N}(\mu_e; \mu_1, \sigma_1^2 + \sigma_e^2), \qquad (11)$$

if $\sigma_3^2 > \sigma_2^2$. The parameters μ_e, σ_e^2 and the constant C are given by:

$$\mu_e = \frac{\mu_2 \sigma_3^2 - \mu_3 \sigma_2^2}{\sigma_3^2 - \sigma_2^2} \qquad (12)$$

$$\sigma_e^2 = \frac{\sigma_2^2 \sigma_3^2}{\sigma_3^2 - \sigma_2^2} \qquad (13)$$

$$C = \frac{\mathcal{N}(0; \mu_2, \sigma_2^2)}{\mathcal{N}(0; \mu_3, \sigma_3^2)\mathcal{N}(0; \mu_e, \sigma_e^2)} \qquad (14)$$

Identifying $\mathcal{N}(x; \mu_1, \sigma_1^2)$ with the Gaussian densities of the acoustic model $p(\mathbf{x}_t | \mathbf{s}_t)$, $\mathcal{N}(x; \mu_2, \sigma_2^2)$ with the a posteriori density of the sent vector and $\mathcal{N}(x; \mu_3, \sigma_3^2)$ with its a priori density, we reduced the integration to computing the probability of observing the feature μ_e given an adapted acoustic model whose variance is increased by σ_e^2. The extension to a Gaussian mixture model for $p(\mathbf{x}_t | s_t)$ is straight forward.

4. Experimental results

This section presents the results of the test we performed in order to evaluate the improvement obtained by considering the a priori term, on one hand, and the modeling with extended state space on the other. We took the reconstruction based on nearest frame repetition (NFR) employed in [5] as a reference.

We simulated a packet-oriented transmission where each packet consisted of two feature vectors. The losses have been induced by a 2-states Markov chain [10], characterized by the conditional loss probability clp and mean loss probability mlp. The set of investigated conditions are summarized in the Table 2.

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

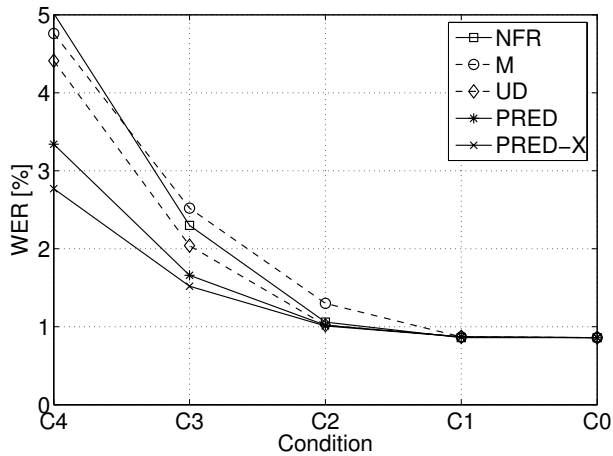| Condition	C1	C2	C3	C4
clp	0.147	0.33	0.5	0.6
mlp	0.006	0.09	0.286	0.385

Figure 2: *Word Error Rates vs. channel condition.*

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 [11].

The ETSI advanced front-end for DSR [5] was employed for feature extraction and quantization. The word error rate in the error free scenario was 0.86% for this setup. To build the Markov model with extended state space, we quantized the dynamic components of each feature subvector with 3 bits for delta and 1 bit for acceleration.

Figure 2 shows the word error rates versus channel condition for following error concealment schemes:

- NFR: the Nearest Frame Repetition scheme

- M: Marginalization of lost frames

- UD: Uncertainty Decoding, i.e. ignoring the a priori term $p(\mathbf{X})$ in (4) and source modeled as in section 3.1.2

- PRED: same as UD but employing the predictive decision rule (4)

- PRED-X: predictive decision rule and extended source model from section 3.1.3

If the loss bursts are relatively short (C2, C3), repetition performs slightly better than marginalization. This is actually expected due to short term correlation, see Table 1. However if the burst increases in duration, it is better to marginalize. Uncertainty decoding (UD) performs better than NFR and M due to its capability to deemphasize the contribution of unreliable features. A significant boost in accuracy is given by considering the a priori term $p(\mathbf{X})$ in Equation (4). The last curve shows another 20% relative gain over PRED which is obtained by the improved source modeling.

Altogether it is a considerable reduction of WER by almost 50% relative to the baseline standard (NFR) in bad channel conditions (C4).

5. Conclusions

In this paper the Bayesian framework of speech recognition was reformulated for the server side of a distributed system. This resulted in a predictive decision rule which was experimentally proven to be more robust against channel errors than the existing methods based on uncertainty decoding.

By modeling the correlation among successive static and dynamic feature vector components, the residual inter-frame redundancy, which is the only source of information to reconstruct the lost frames, is more effectively exploited.

7. 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] 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.

[3] J. Aitchinson and I.R. Dunsmore, *Statistical Prediction Analysis*, Cambridge University Press, 1975.

[4] M. Cooke, P. Green, L. Josifovski, and A. Vizio, "Robust automatic speech recognition with missing and unreliable acoustic data," *Speech Communication*, vol. 34, pp. 267–285, 2001.

[5] ES.202.050, "Speech processing, Transmission and Quality aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms," *ETSI*, Oct 2002.

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

[7] 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.

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

[9] V. Ion and R. Haeb-Umbach, "Uncertainty decoding for distributed speech recognition over error-prone networks," *Submitted to Speech Comm., Special issue on Robustness Issues in Conversational Interaction*.

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

[11] 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.