Joint Channel Decoding - Viterbi Recognition
For Wireless Applications

Alexis Bernard and Abeer Alwan

Dept. of Electrical Engineering, University of California, Los Angeles

{abernard, alwan}@icsl.ucla.edu

Abstract

We introduce the concept of joint channel decoding and Viterbi recognition, by which the Viterbi recognizer is modified to take into account the confidence in the decoded feature after channel transmission. We present a metric for evaluating such confidence based on soft decision decoding. As a case study, we quantize MFCCs using predictive VQ. The overall source-channel coding scheme operating at a combined rate of 1 kbps is shown to provide good recognition accuracy over a wide range of Rayleigh fading channels.

1. Introduction

In this paper, we investigate source and channel coding, and recognition techniques suitable for wireless distributed speech recognition. The goal is to provide high recognition accuracy over a wide range of channel conditions with low bitrate, delay and complexity for the client.

Recent papers addressing the issue of quantizing speech recognition features include [1, 2, 3, 4, 5, 6]. In [1][2], the line spectral pairs (LSP) of the Perceptual Linear Prediction (PLP) coefficients are quantized, taking advantage of the low-dimensionality of the PLP feature vector and of the quantization properties of the LSPs. In [3], linear prediction speech coders parameters such as LSPs and gains are used to compute speech recognition features. In [4], split Vector Quantization (VQ) of Mel-Frequency Cepstral Coefficients (MFCC) is shown to provide good recognition accuracy at about 2 kbps. [5] uses similar techniques to provide recognition at 4 kbps. [6] exploits redundancy of MFCC parameters using a 2-D Discrete Cosine Transform (DCT).

In most wireless applications, the transmission channel is error-prone and quantized features must be protected against transmission errors. Previous studies have suggested alleviating the effect of channel errors by adapting HMM models [7] and ASR front-ends [8] to different channel conditions, or by modeling GSM noise and holes [9]. Other studies analyzed the effect of random and burst errors in the GSM bitstream for remote speech recognition applications [10]. Recently, [1] introduced channel coding techniques for remote recognition with soft decoding. It is also shown that speech recognition, in contrast with speech coding, can be more sensitive to channel errors than channel erasures.

The contribution of this paper is multi-fold. First, the Viterbi recognizer is modified to include a time-varying weighting factor depending on the reliability of each decoded feature. Second, a technique for computing the reliability based on the soft received bits is presented. Third, an efficient MFCC quantization scheme operating at 500-900 bps is presented; the quantized features are used as a case study for the modified Viterbi

recognizer. Fourth, channel codes for error detection are proposed. The source-channel coding scheme operating at a combined rate of 1 kbps is shown to provide high recognition accuracy over a wide range of channel conditions.

2. Channel errors and speech recognition

2.1. The Viterbi algorithm for speech recognition

Speech recognition is performed by selecting the element in the dictionary that is the most likely to produce a sequence of observations. The likelihood of observing a given sequence of features given a Hidden Markov Model (HMM) is computed by searching through a trellis for the most probable state sequence. The Viterbi Algorithm (VA) presents a dynamic programming solution to find the most likely path through a trellis (Fig. 1). For each state j, at time t, the likelihood of each path is computed by multiplying the transition probabilities a_{ij} between states and the output probabilities $b_j(o_t)$ along that path. The partial likelihood $\sigma_{j,t+1}$ is computed efficiently using the following recursion

$$\sigma_{j,t+1} = \max_i[\sigma_{i,t}\ a_{ij}]\ b_j(o_t). \tag{1}$$

The probability of observing the N_F-dimensional feature o_t is

$$b_j(o_t) = \sum_{m=1}^{N_M} c_m \frac{1}{\sqrt{(2\pi)^{N_F}|\Sigma|}} e^{-\frac{1}{2}(o_t-\mu)'\Sigma^{-1}(o_t-\mu)} \tag{2}$$

where N_M is the number of mixture components, c_m is the mixture weight and the parameters of the multivariate Gaussian mixture are its mean vector μ and covariance matrix Σ.

2.2. Effects of channel transmission on speech recognition

In remote speech recognition, especially in wireless communication where fadings occur, the decoded feature is a function of the transmission channel characteristics. When channel characteristics degrade, one can no longer guarantee the reliability of the decoded feature. If the VA operates without taking into account the decreased confidence in the feature, this can have a dramatic effect on speech recognition accuracy since maximum likelihood trellis searches accumulate metrics over time and errors in decoding a feature will propagate in the path metrics.

Throughout this paper, speech recognition experiments consist of continuous digit recognition based on 4 kHz bandwidth speech signals. Training is done using speech from 55 males and females from the Aurora-2 database for a total of 2200 digit strings. Word HMM models in the Aurora configuration contain 16 states and 6 mixtures and are trained using the Baum-Welch algorithm assuming a diagonal covariance matrix. Recognition tests contain 1000 digit strings spoken by 100 different speakers (male and female) for a total of 3241 digits. Recognition results are reported as word accuracy.
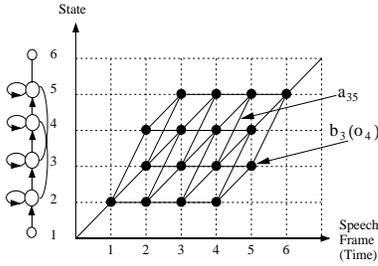
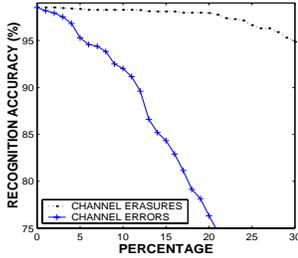Figure 1: *Viterbi Algorithm visualization (after[11]).*

Figure 2: *Effect of channel erasures and channel errors on word-model based continuous digit recognition.*

Fig. 2 illustrates the effect of channel erasures and channel transmission errors on the recognition accuracy of a word-model based continuous digit recognition experiment. Note that channel errors, which propagate through the trellis search, have a disastrous effect on recognition accuracy while the recognizer is still able to operate with almost no loss of accuracy with up to 15% of channel erasures. This confirms results obtained in [1] for isolated digit recognition based on PLP coefficients.

2.3. Channel-matched Viterbi recognition

In this paper, we present a solution for modifying the recursive step (1) of the VA so as to take into account the effect of transmission errors. Ideally, one would like to weight the probability of observing the decoded feature given the HMM state model $b_j(\boldsymbol{o}_t)$ with the probability of decoding the feature vector $\boldsymbol{o}_t$ given the received bit values $\boldsymbol{y}_t$. The time-varying weighting coefficient $\gamma_t = p(\boldsymbol{o}_t|\boldsymbol{y}_t)$ can be inserted into (1) to obtain

$$\sigma_{j,t+1} = \max_i \left[\sigma_{i,t} \, a_{ij} \right] \left[b_j(\boldsymbol{o}_t) \right]^{\gamma_t}. \tag{3}$$

Note that if one is certain about the received feature (no noise), γ_t is equal to one and (3) is equivalent to (1). On the other hand, if the decoded feature is completely unreliable, $\gamma_t = 0$ and the probability of observing the feature given the HMM state model $b_j(\boldsymbol{o}_t)$ is discarded in the VA recursive step.

Finally, note that the hypothesis of diagonal covariance matrix $\boldsymbol{\Sigma}$ in (2) is often made for MFCCs since the feature vector is obtained after a DCT decorrelating operation. Consequently, (2) can be computed as the product of the probabilities of observing each individual feature. If the features are quantized and transmitted separately, the channel-matched recursive formula (3) is improved to include individual weighting factors $\gamma_{k,t}$ for each of the N_F features,

$$\sigma_{j,t+1} = \max_i \left[\sigma_{i,t} \, a_{ij} \right] \prod_{k=1}^{N_F} \left[b_j(o_{k,t}) \right]^{\gamma_{k,t}}. \tag{4}$$

2.4. Estimating the decoding confidence factor γ

For a discrete memoryless channel, the probability of receiving the vector $\boldsymbol{y}$ (N bits) given that the codeword $\boldsymbol{x}_m$ (K bits) was transmitted is given by

$$p(\boldsymbol{y}|\boldsymbol{x}_m) = \prod_{j=1}^{N} p(y_j|x_{mj}) \qquad (0 \le m \le 2^K - 1) \tag{5}$$

which can be re-written as

$$p(\boldsymbol{y}|\boldsymbol{x}_m) = \left(\frac{1}{\sqrt{\pi N_0}} \right)^N \exp\left[-\sum_{j=1}^{N} \frac{(y_j - x_{mj})^2}{N_0} \right] \tag{6}$$

for a soft decision memoryless channel (N_0 is the average noise energy). For a binary symmetric channel (BSC) with cross-over probability p or for any channel with hard decision decoding $p(\boldsymbol{y}|\boldsymbol{x}_m) = p^{d_H}(1-p)^{N-d_H}$ where $d_H = d_H(\boldsymbol{y}, \boldsymbol{x}_m)$ is the Hamming distance between $\boldsymbol{y}$ and $\boldsymbol{x}_m$. For a Rayleigh fading channel, $p = Q(\sqrt{2\alpha^2 E_b/N_0})$ where E_b is the average energy per bit and α is a Rayleigh distributed random variable.

For any channel encoder, the optimal decoding rule is to pick $\hat{\boldsymbol{x}} = \boldsymbol{x}_m$ that maximizes $p(\boldsymbol{y}|\boldsymbol{x}_m)$. This is equivalent to minimizing the Euclidean distance $d_E = \sum_{j=1}^{N} (y_j - x_{mj})^2$ between $\boldsymbol{y}$ and $\boldsymbol{x}_m$ for soft decision decoding, or minimizing the Hamming distance d_H between $\boldsymbol{y}$ and $\boldsymbol{x}_m$ for hard decision decoding.

In summary, the decoder selects the closest codeword $\boldsymbol{x}_m$ (with respect to the Euclidean or Hamming distances) to the received codeword $\boldsymbol{y}$. The remaining question is to evaluate how confident we are about this decision or, equivalently, what is the probability $p(\hat{\boldsymbol{x}} = \boldsymbol{x}_m|\boldsymbol{y})$. Using Bayes rule and assuming that all codewords are equiprobable, this can be evaluated as follows

$$p(\hat{\boldsymbol{x}} = \boldsymbol{x}_m|\boldsymbol{y}) = \frac{\prod_{j=1}^{N} \exp[-\frac{(y_j - x_{mj})^2}{N_0}]}{\sum_{m'=0}^{2^K-1} \prod_{j=1}^{N} \exp[-\frac{(y_j - x_{m'j})^2}{N_0}]} \tag{7}$$

for soft decision decoding and as

$$p(\hat{\boldsymbol{x}} = \boldsymbol{x}_m|\boldsymbol{y}) = \frac{p^{d_H(\boldsymbol{y},\boldsymbol{x}_m)}(1-p)^{N-d_H(\boldsymbol{y},\boldsymbol{x}_m)}}{\sum_{m'=0}^{2^K-1} p^{d_H(\boldsymbol{y},\boldsymbol{x}'_m)}(1-p)^{N-d_H(\boldsymbol{y},\boldsymbol{x}'_m)}} \tag{8}$$

for hard decision decoding or BSC channels.

Note that (7) and (8) are complex and require the knowledge of N_0. The noise variance can sometimes be evaluated using complex channel state information tracking or probing techniques. In this paper, we present a solution for estimating the confidence in the decoding of the feature based on the relative difference between the two closest valid codewords from the received bit sequence.

The codeword $\hat{\boldsymbol{x}}$ exhibiting the smallest Euclidean distance d_{E_1} with the received vector $\boldsymbol{y}$ is decoded. To evaluate confidence in the decoding decision, the second smallest Euclidean distance d_{E_2} between $\boldsymbol{y}$ and the codewords $\boldsymbol{x}_m$ is also computed. A good measure for the reliability of the decoding rule is the relative difference β between both distances

$$\beta = \frac{d_{E_2} - d_{E_1}}{d_{E_1}}. \tag{9}$$

If the received vector $\boldsymbol{y}$ lies exactly between two valid codewords, $\beta = 0$, and the decoder's best decision is a guess between both codewords. On the other hand, if there is no noise in the channel, $d_{E_1} = 0$ and $\beta = \infty$. This shows that β can be used as a confidence measure of the decoding operation. Note also that (9) is independent of the channel noise N_0.
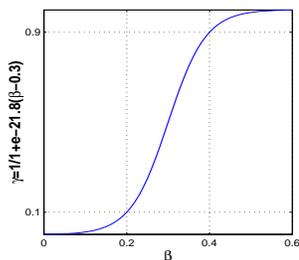
Figure 3: *Sigmoid function mapping relative Euclidean distance difference (β) to confidence measure (γ).*

SNR	E	C_1	C_2	C_3	C_4	C_5	C_6
-5 dB	7.4	19.1	33.9	26.1	43.8	60.5	72.5
0 dB	10.3	61.1	72.1	88.4	89.1	93.6	95.9

SNR		C_7	C_8	C_9	C_{10}	C_{11}	C_{12}
-5 dB		55.9	59.9	72.1	75.4	88.8	86.5
0 dB		96.0	96.5	96.9	97.2	97.6	97.6

Table 1: *Recognition accuracy after quantization noise (SNR dB) is added to each feature.*

bits/sec.	bits/frame	MFCC_E	MFCC
Unquantized	—	98.46	97.28
900	9+9	98.30	97.17
800	8+8	98.08	97.07
700	7+7	97.56	96.30
600	6+6	97.32	95.32
500	5+5	96.88	93.45

Table 2: *Recognition accuracy after quantizing the MFCCs using first order predictive weighted split VQ. Notation 8+8 means 8 bits for the first split and 8 bits for the second.*

This metric is used in [1] to perform soft decision error-detection, crucial for speech recognition. When $\beta < 0.2$, the feature vector is declared in error and the corresponding frame is dropped. This means that if there are any two valid codewords within a sphere of radius $1.2 \cdot d_{E1}$ of the received codeword $\boldsymbol{y}$, the decoder is likely to make a mistake and the frame is erased. The soft decision channel decoder, which introduces additional complexity only at the server, is proven to outperform hard decision decoding for both error correction and error detection.

In [1], the binary decision of dropping the frame is done before recognition. In this paper, the system is refined by transmitting the coded feature to the Viterbi recognizer, along with the confidence γ_t in the decoded feature. This presents three advantages over the solution presented in [1]. First, the time resolution in the state sequence estimation is unaltered since the state metrics are still updated in (3) using a_{ij}, even if $b_j(\boldsymbol{o}_t)$ is discarded. Second, we can develop a mapping function between the decoding measure β ($0 \leq \beta \leq \infty$) and the Viterbi weighting coefficient γ_t ($0 \leq \gamma_t \leq 1$) in order to use the channel-matched Viterbi recursive step (3). We propose the following sigmoid function,

$$\gamma_t = 1/(1 + e^{-21.8(\beta - 0.3)}) \qquad (10)$$

to map the relative difference in Euclidean distances β into confidence estimate γ. This function, shown in Fig. 3, gives a confidence measure $\gamma < 0.1$ when $\beta < 0.2$ and $\gamma > 0.9$ when $\beta > 0.4$. Third, individual weighting $\gamma_{k,t}$ for each feature can be computed within a frame and utilized as in (4).

3. Case study: Transmission of MFCCs

3.1. Efficient MFCC quantization

Typically, a speech recognition feature vector consist of 12 MFCCs ($C_1, \ldots, C_{12}$), to which might be added a log-energy component ($\log(E)$). MFCCs are computed every 10 ms using a 25 ms analysis window. This overlap results in high correlation between adjacent frames. This correlation can be exploited in distributed speech recognition systems. Specifically, the client can compute and transmit features every 20 ms, while the server interpolates the features by a factor of 2 for recognition. This results in lower bitrate and complexity at the client.

Furthermore, due to the nature of the speech signal itself, there is evidence of remaining correlation between adjacent frames even if MFCCs are computed every 20 ms. This is captured in our coding scheme using first order predictive coding, which provides on average 4 dB of coding gain. The MFCCs can then be efficiently quantized as follows: 1) remove the mean of each feature; 2) compute the residual feature after first order

linear prediction whose coefficient is chosen to minimize the signal variance after prediction; 3) split the residual vector into 2 subvectors and vector quantize them using different rates depending on channel conditions. Note that the cost function to be minimized during VQ training and VQ search is weighted to take into account quantization sensitivities of each feature. Degradations in recognition when quantization noises at different SNRs are added to each feature are shown in Table 1.

Table 2 reports continuous digit recognition accuracy when quantizing MFCCs (with and without energy) at different bitrates for each VQ split. When using MFCC without energy, the residual is split [C_1-C_6] and [C_7-C_{12}]. If energy is added (MFCC_E), the feature vector is split [$\log(E)$,C_1-C_5] and [C_6-C_{12}]. Training and testing are done on quantized features. Note that one can get good recognition accuracies with rates as low as 500 bps. Below this, however, prediction starts degrading and recognition drops significantly. Note also that despite the additional vector dimension to quantize, MFCC_E always outperforms MFCC.

3.2. Efficient channel coding for error detection

Depending on channel conditions, the effect of channel noise on the received bits can be so that for most feature vectors, the confidence measure is almost zero most of the time. This is the case under severe noise condition (or fading) and if all 2^K valid codewords are close to each other in the N-dimensional space of the received bit sequences.

The role of channel coding is to map K bits representing a feature vector into N bits ($N > K$) in such a way as to maximize the minimum distance between valid codewords in the N-dimensional space. We refer to the resulting code as an (N, K) code. The larger the redundancy ($N - K$), the larger the minimum distance (d_{min}) between any two valid codewords. Since the number of source information bits necessary to code each frame is small, block codes are favored over convolutional or trellis codes for delay and complexity considerations.

In order to guarantee good recognition accuracy over a wide range of channel conditions, different block codes with different

(N,K)	d_{min}	A_0	A_1	A_2	A_3	A_4	A_5
(10,9)	2	1	0	45	0	210	0
(10,8)	2	1	0	12	36	46	60
(10,7)	2	1	0	3	19	29	27
(10,6)	3	1	0	0	9	16	15
(10,5)	4	1	0	0	0	16	0

Table 3: *Distance spectra of the linear block codes used for channel coding of ASR features.*
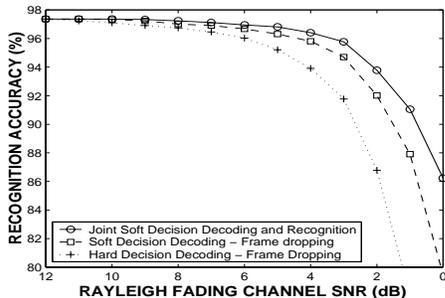

Figure 4: *Recognition accuracy using MFCC_E and the (10,6) linear block code over an independent Rayleigh fading channel.*

detection capabilities are used. With such a scheme that adapts source and channel coding rates depending on channel conditions, graceful degradation in recognition performance is provided with decreasing channel quality while keeping the overall bitrate constant.

In this paper, the overall (source and channel) operating bitrate is set to be 1 kbps (20 bits/frame or 10 bits/split). The block codes used range from $(10, 9)$ to $(10, 5)$. The $(10, 5)$ code is obtained by expurgating the odd-weight codeword from a $(15, 11)$ Hamming code ($d_{min} = 3$) to form a $(15, 10)$ code that is then shortened to give a $(10, 5)$ code. The $(10, 6)$ code is obtained by shortening the same $(15, 11)$ Hamming code. The codes $(10, 7)$ and $(10, 8)$ are obtained by finding the best combination for expurgating and puncturing the $(15, 11)$ Hamming code. Finally, the code $(10,9)$ is a simple Cyclic Redundancy Check (CRC) code. The partial distance spectrum of each code is presented in Table 3.

3.3. Results over an independent Rayleigh fading channel

Fig. 4 compares recognition accuracy for joint soft channel decoding - Viterbi recognition introduced here, soft decision channel decoding with error detection introduced in [1] and the widely used hard decision channel decoding when using the (10,6) linear block code over an independent Rayleigh fading channel. In the last two scenarios, frames are dropped if channel errors are detected. Note that performing joint channel decoding and recognition has a clear advantage.

Fig. 5 illustrates recognition accuracy after choosing for each SNR the block code that yields the best results. Note again the superior performance of the joint soft decision decoding - Viterbi recognition scheme.

4. Summary

We have introduced the concept of joint channel decoding and Viterbi recognition, by which the Viterbi recognizer is modified
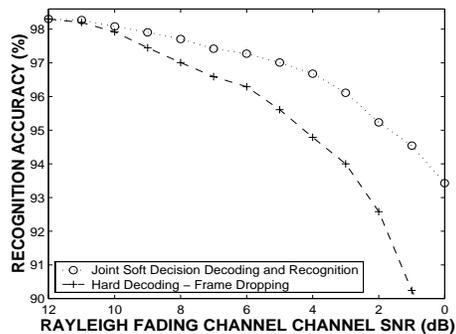

Figure 5: *Recognition accuracy after transmission of MFCC_E over an independent Rayleigh fading channel.*

to take into account the confidence in the decoded feature after channel transmission. We presented a metric for evaluating such confidence based on soft decision channel decoding. As a case study, we quantized MFCCs using predictive VQ at rates from 500 to 900 bps. The overall source-channel coding scheme operating at 1 kbps is shown to provide good recognition accuracy over a wide range of Rayleigh fading channels.

5. References

[1] A. Bernard and A. Alwan, "Source and channel coding for remote speech recognition over error-prone channels," in *Proc. of ICASSP 2001*, to appear.

[2] W. Gunawan and M. Hasegawa-Johnson, "PLP coefficients can be quantized at 400 bps," in *Proc. of ICASSP 2001*, to appear.

[3] H.K. Kim and R. Cox, "Bitstream-based feature extraction for wireless speech recognition," in *Proc. of ICASSP*, 2000, vol. 3, pp. 1607–10.

[4] V. Digalakis, L. Neumeyer, and M. Perakakis, "Quantization of cepstral parameters for speech recognition over the www," *IEEE J. Sel. Areas Comm.*, pp. 82–90, Jan. 1999.

[5] G. Ramaswamy and P. Gopalakrishnan, "Compression of acoustic features for speech recognition in netword environments," in *Proc. of ICASSP*, 1998, vol. 2, pp. 977–80.

[6] Q. Zhu and A. Alwan, "An efficient and scalable 2D-DCT based feature coding scheme for remote speech recognition," in *Proc. of ICASSP 2001*, to appear.

[7] T. Salonidis and V. Digalakis, "Robust speech recognition for multiple topological scenarios of the GSM mobile phone system," in *Proc. of ICASSP*, 1998, pp. 101–4.

[8] S. Dufour, C. Glorion, and P. Lockwood, "Evaluation of the root-normalized front-end for speech recognition in wireless GSM network environments," in *Proc. of ICASSP*, 1996, vol. 1, pp. 77–80.

[9] L. Karray, A. Jelloum, and C. Mokbel, "Solutions for robust recognition over the GSM cellular network," in *Work. Interac. Voice Techn. for Telec. Appl.*, 1998, pp. 166–170.

[10] A. Gallardo, F. Diaz, and F. Vavlerde, "Avoiding distortions due to speech coding and transmission errors in GSM ASR tasks," in *Proc. of ICASSP*, 1999, pp. 277–80.

[11] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.0)*, July 2000.