ABSTRACT

In large-vocabulary speech recognition previous knowledge about the voice of the speaker is extremely useful for achieving high accuracy. For systems based on VQ and HMMs, a training sample is collected from the new speaker to build the VQ codebook and HMM parameters. We describe some techniques aimed at achieving a fast adaptation of speaker-independent codebook and HMM parameters, based on a small speech sample provided by the new speaker. Results are presented concerning the 20000-word vocabulary, isolated-utterance, real-time IBM recognizer for the Italian language.

INTRODUCTION

In large-vocabulary speech recognition speaker-dependent parameters allow high accuracy [1]. Each new user is required to utter a training text; then, a (usually long) computation procedure is performed on the sample data, in order to find the optimal parameters. While this is not a major problem for many applications, in some cases a shorter enrollment time would be desirable.

The 20000-word isolated-utterance real-time IBM prototype recognizer for the Italian language [2][3] requires a 15-minute training speech sample from each new speaker. The sample is used for two purposes:
- Finding a 200-element codebook, to be employed for vector quantization, via k-means clustering.

Because the training process requires a substantial amount of computation, the trained recognizer is available to the new user only after a few hours. The word recognition accuracy achieved is above 95%. If speaker-independent parameters are used instead, the system is available immediately to any user, but accuracy may become as low as 80%.

The work described here is aimed at approaching the accuracy of speaker-trained recognition by a fast adaptation of the codebook and of HMM parameters based on a small speech sample provided by the new speaker.

Previous works on codebook adaptation are usually based on techniques for modifying the codebook of a reference speaker in order to reduce the vector-quantizer distortion for the new speaker [4]. An unsupervised spectral-mapping scheme is described in [5].

For HMM parameter adaptation, the most popular approach consists in computing a confusion matrix which relates the codewords output by the reference and by the new vector quantizer [6][7][8]. The use of "Speaker Markov Models" has allowed to reach a recognition accuracy close to that of fully trained HMMs using only a 5-minute long speech sample [9].

For codebook adaptation, we developed two techniques which make use of a priori knowledge (acquired from speech samples of several speakers) to estimate the statistical significance of the adaptation sample. The two techniques, based respectively on Bayesian estimation and on deleted estimation, are outlined in [10] and [11].

The application of Bayesian learning to VQ prototype adaptation, together with a technique based on tied-mixture continuous-parameter Markov models, is also described in [12].

The present paper is organized as follows. The next section contains a brief description of our approach to codebook adaptation. The following section introduces a technique for HMM parameter adaptation and presents some preliminary experimental results. The last section outlines future developments of our work.

CODEBOOK ADAPTATION

Two approaches were investigated to overcome the problem of statistical insufficiency of the new speaker's speech sample S: Bayesian estimation and deleted Euclidean interpolation [10][11]. In both cases common prototypes, obtained by standard K-means clustering on speech collected from 10 different speakers, are used as reference.
The first approach to prototype adaptation is based on Bayesian estimation. The set of vectors assigned to each prototype is modeled by a diagonal multivariate Gaussian probability distribution, of which the prototype is the mean. Let \( \mu \) be the mean of a component and let \( \sigma^2 \) be its variance (assumed to be known). We look for the value of \( \mu \) which maximizes the a posteriori probability distribution:

\[
P(\mu | X) = \frac{P(X | \mu)P(\mu)}{P(X)}
\]

where \( X \) is the observed data, \( P(X | \mu) \) is the conditional probability of the data given the mean and \( P(\mu) \) is the a priori probability of the data. The a priori probability distribution of the mean, \( P(\mu) \), is also assumed to be Gaussian, with mean \( \mu_0 \) and variance \( \sigma^2_0 \). The value of \( \mu \) which maximizes the a posteriori probability is [13]:

\[
\mu = \frac{n \sigma^2_0 \mu_0 + a^2}{n \sigma^2_0 + a^2}
\]

where \( n \) is the number of elements of sample \( X \) and \( m_n \) is the sample average. Sample \( X \) for the \( k \)-th prototype is obtained from the speech sample by taking all vectors labeled with the \( k \)-th common prototype. \( \mu_k \) is estimated as the common prototype; \( \sigma^2 \) as the averaged squared distance of speaker-specific prototypes from common prototypes; \( \sigma_0^2 \) as the averaged squared distance of speaker-specific prototypes from corresponding speaker-produced vectors.

The second approach looks for the optimal (in the Euclidean sense) linear interpolation between the common prototypes \( C_k \) and the prototypes \( S_k \), obtained from \( S \) by computing the vector means. A deleted estimation technique is employed [14]. The \( k \)-th component of the adapted prototype \( A_k \) is given by

\[
A_k = \lambda_k C_k + (1 - \lambda_k) S_k
\]

where \( \lambda \) indicates a bin dependent on the amount of data available for prototype \( k \) in \( S \). \( \lambda_k \) is estimated by minimizing total distortion, i.e. the average Euclidean distance of the vectors of sample \( S \) from their corresponding adapted prototypes.

Both techniques allow a very fast computation of the adapted prototypes, much less computationally expensive than standard K-means clustering.

Decoding tests were performed after a complete training of the HMM parameters from sample 1. The following table (already reported in [11]) shows percent recognition accuracy for four speakers, using clustered (from sample 1), common and adapted (by procedure 1 and 2 respectively) prototypes.

<table>
<thead>
<tr>
<th>Spk</th>
<th>CLU</th>
<th>COM</th>
<th>ADP1</th>
<th>ADP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>98.0</td>
<td>95.7</td>
<td>98.0</td>
<td>97.7</td>
</tr>
<tr>
<td>2</td>
<td>95.7</td>
<td>90.0</td>
<td>95.7</td>
<td>95.4</td>
</tr>
<tr>
<td>3</td>
<td>96.1</td>
<td>93.8</td>
<td>94.2</td>
<td>94.2</td>
</tr>
<tr>
<td>4</td>
<td>96.1</td>
<td>95.2</td>
<td>97.7</td>
<td>97.7</td>
</tr>
</tbody>
</table>

For HMM parameter estimation, a short adaptation text is uttered by the speaker. We use the deleted estimation technique to interpolate different statistics:

- Speaker-dependent statistics for each phone; they are computed by the forward-backward algorithm, from the adaptation text uttered by the new speaker.
- Speaker-dependent statistics, computed in the same way as from the same data, for classes of phones obtained tying together acoustically similar phones.
- Speaker-independent statistics for each phone obtained from large sample provided by several speakers.

Speaker-independent parameters are also used as initial guesses in the forward-backward iteration.

To find classes of phones, a hierarchical classification is performed. Several techniques for this purpose are described in [15], where the are applied to the problem of fast lexical access. For speech adaptation, we performed a clustering based on the divergence of the output probabilities of the phones. 14 classes were found for phonetic phones: 35 for phonemic [16] phones.

Only very preliminary experimental results are available at this tim

One speaker provided a 5-minute long adaptation sample, by reading a phonetically-balanced text A consisting of 30 sentences, amounting to 345 words. Word recognition accuracy was measured on disjoint text, consisting of 61 sentences amounting to 1024 word uttered by the same speaker. In all experiments, the VQ prototypes were obtained by standard K-means clustering on a large sample (that is, no fast adaptation of the prototypes was performed).

The following table gives the accuracy achieved in three different conditions:

<table>
<thead>
<tr>
<th>HMM Adaptation</th>
<th>Percent Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>L-ST</td>
<td>97.6</td>
</tr>
<tr>
<td>A-ST</td>
<td>93.2</td>
</tr>
<tr>
<td>A-AT</td>
<td>95.7</td>
</tr>
</tbody>
</table>

Table 2. Recognition accuracy for different HMM parameters.
FUTURE DEVELOPMENTS

While the techniques for adaptation we described are promising, much work is still needed in this area. More experiments on several speakers have to be performed. The adaptation of the codebook will be combined with that of the HMM parameters. The performance of the adaptation as a function of the amount of data will also be studied. Codebook adaptation is already very fast; techniques for reducing the computation required by HMM adaptation will be addressed.

REFERENCES


