

Low Memory Acoustic Models for HMM Based Speech Recognition

Tommi Lahti, Olli Viikki, Marcel Vasilache

Speech and Audio Systems Laboratory

Nokia Research Center, Tampere, Finland

{tommi.lahti, olli.viikki, marcel.vasilache}@nokia.com

Abstract

In this paper, we propose a new approach to reduce the memory footprint of HMM based ASR systems. The proposed method involves three steps. Starting from the continuous density HMMs, mixture variances are tied using k -means based vector quantization. Next, the re-estimation of the resulted models is performed with tied variances. Finally, scalar quantization is performed for the mean components of the models. With the proposed method, a memory saving of 77.6% was achieved compared with the original continuous density HMMs and 23.0% compared to the quantized parameter HMMs, respectively. The recognition performance of the resulted models was similar to what was obtained with the original continuous density HMMs in all tested environments.

1. Introduction

To date, acoustic modeling in Automatic Speech Recognition (ASR) is commonly based on the Continuous Density Hidden Markov Models (CDHMM) with Gaussian mixture densities. While the basic CDHMM framework can be well utilized in several ASR systems, there are certain ASR application areas, e.g. embedded implementation platforms, where special optimizations are required in order to take advantage of the HMM technology. In particular, the memory requirements of the HMMs and the computational complexity of evaluating the observation probabilities are of high importance.

During the recent years, many techniques have been developed to reduce the memory footprint and computational complexity of HMMs. Some of these methods are also capable of improving the recognition performance. One such class of methods is based on various ways of parameter tying. For example, mixture or state level tying schemes in Large Vocabulary Continuous Speech Recognition (LVCSR) are usually aimed for more robust estimation of parameters through utilization of increased training data as well (see [1] and the references therein).

Vector Quantization (VQ) has been utilized for memory reduction in [2] and [3]. In [2], the codebook was designed to minimize the total distortion between the centroid code-vectors and the original vectors. In

[3], the quantization of the variance vectors was based on the information theoretic distortion measure. Methods for multilingual acoustic modeling have been investigated in [4].

In [5], it was shown that scalar quantization even with a fairly low number of quantization levels for mean and variance values has no significant effect on the recognition performance.

In this paper, we combine three different techniques in order to reduce the memory footprint of the acoustic models and computational complexity. Starting from the conventional CDHMMs (baseline), vector quantization is performed by tying the variance parameters of Gaussian mixtures. Then, a few re-estimation rounds with tied variances are carried out. As a final step, scalar quantization is performed to further reduce the memory consumption. The method also inherits the computational advantages from the methods it is utilizing.

The remainder of the paper is organized as follows. In Section 2, combining of three methods for memory and computational complexity reduction is discussed. Memory and computational complexity considerations are made in third section. Experimental results are given in Section 4. Finally the conclusions can be found in Section 5.

2. Compressed models representation

In the literature, several model size reduction techniques have been proposed. In many cases, computational overhead can also be reduced. Many of these methods are more or less independent of each other so they can be applied jointly to optimize the implementation. In this paper, we will combine three such techniques namely vector quantization, parameter tying and scalar quantization.

2.1. Vector quantization and parameter tying

It has been shown in several studies that the mean and variance vectors of the CDHMMs can be quantized without significant performance degradation (see for example [2] and [3]). As a result, less actual mean or/and variance vectors are needed to be stored resulting in significant memory savings. By converting observation probability computations into table lookup

operations, computational overhead can also be reduced.

Vector quantization could be seen as a way of deriving parameter tying. However, parameter-tying techniques may also contain the Maximum Likelihood (ML) re-estimation step of the HMMs, which is missing from the pure VQ approach. In addition to the memory savings and the computational complexity reduction, tying of the model parameters has also other advantages. As more training data is now available, the modeling accuracy may become better and the recognition rate increases. This is especially true with context dependent models and LVCSR. It is also well known that parameter tying (especially tying of the variances) can improve the recognition performance in noise [6].

In the pure VQ approach, the codebook needs to be well optimized because of the lack of re-estimation. For example, the classical k -means algorithm may be far from the optimum as pointed out in [2] and [3]. In the proposed approach, the optimization of the codebook is not so crucial. After the VQ step, the tied variance vectors are marked and a couple of ML re-estimation rounds with these tied parameters are carried out. From the parameter estimation point of view, the resulted models are again optimized in the ML sense. It should be noted that quantization of the mean vectors is not needed in the proposed approach. The reason is that the mean vectors are less tolerant for VQ and hence no big memory savings could be obtained. Quantization techniques can still be applied as discussed next.

2.2. Scalar quantization - qHMMs

In [5], a scalar quantization technique based on quantized parameter HMMs (qHMM) was successfully utilized to compress the memory footprint of acoustic models. Computational speedup was also attained with the help of table lookups.

In the original paper, two Lloyd-Max scalar quantizers were trained according to the target quantization rates, one for the mean, and one for the variance components of the feature vector, respectively. Hence, a joint quantization scheme was used, as stated in [7]. Next, the conventional CDHMMs were quantized using these quantizers.

It is worth noticing that in principle scalar and vector quantization techniques are independent of each other. However, in the method proposed in this paper, the scalar quantization is applied only for the mean components since the variance vectors have already been heavily quantized. The resulted models can be significantly more compact than the original qHMMs. These models are referred as VQHMMs from this point onwards.

2.3. Putting it all together

The required steps of the proposed method are summarized in Figure 1.

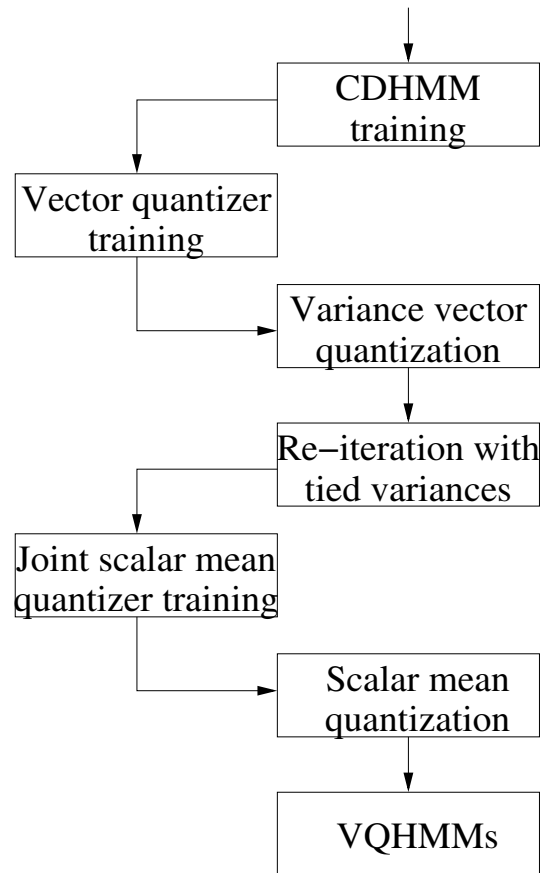


Figure 1: The advance of the proposed training process.

3. Memory and computational complexity considerations

In this work, we only focus on monophone based ASR systems since our primary target platform are embedded devices, such as mobile terminals. The same considerations can though be applied to LVCSR systems. Moreover, we focus on three state left-to-right model structures with a N dimensional acoustic feature vector. The silence model is set to have only one state.

Let us denote the number of density mixtures for the models by D . For the qHMMs, we assume that the number of scalar quantization levels is presented with m bits for the means and with v bits for the variances. For the VQHMMs, in addition to the mean scalar quantization levels, we assume that there are Q quantization levels for variance vectors. The memory figures for the CDHMMs with a different number of densities are presented in the Table 1. The memory consumption of qHMMs and VQHMMs with different

values for D and m with $v=3$, $Q=64$ and $N=39$ are given in Table 2.

Densities	1000	1250	1500	1750	2000
Kilobytes	154	193	231	270	308

Table 1: Memory consumption (kilobytes) of the CDHMMs using a 16 bit floating point representation.

Mixtures	Bits	QHMM	VQHMM	Saving
1000	4	35.3	26.9	24%
	5	40.1	31.7	21%
	6	44.9	36.5	19%
1250	4	44.2	32.4	27%
	5	50.1	38.4	23%
	6	56.1	44.4	21%
1500	4	53.0	37.9	28%
	5	60.1	45.0	25%
	6	67.3	52.2	22%
1750	4	61.8	43.4	30%
	5	70.1	51.7	26%
	6	78.5	60.1	23%
2000	4	70.6	48.9	31%
	5	80.2	58.4	27%
	6	89.7	68.0	24%

Table 2: Memory calculations for the qHMMs and VQHMMs in kilobytes with different number of densities and with the different number of bits for the means. The number of bits used for variance was fixed to three. Floating point presentation was assumed to be with sixteen bits.

Assuming a B bit floating point representation for the mean, variance and mixture weight components, the formula for the memory calculations in bits for CDHMMs is $2NDB + DB$ (naturally, for kilobytes divide this by 8192). For qHMMs, only the mixture weight and the mean and variance codebook components need to be presented with the full B bit presentation. The mean and variance vector components are presented with m and v bits, respectively. Hence, the formula for qHMMs is $B(2^m + 2^v + D) + ND(m + v)$. In addition to the mean codebook, the variance vector codebook is need for the VQHMMs. Variance vectors need to be also assigned to the densities through indexing. If we assume a $B/2$ bit presentation for indexing, the formula for VQHMMs can be written in the form $B(2^m + QN + D) + D(Nm + B/2)$.

It can be seen from the tables above that compared to CDHMMs, the qHMM and VQHMM representations provide substantial memory savings. It can also be seen that VQHMMs provide substantial memory reduction with respect to the qHMMs based acoustic models.

The computational saving of VQHMMs is inherited from qHMMs. Using qHMMs with m bits for mean and v bits for variance, we need only to calculate the B -probability values to the $2^m \times 2^v$ sized lookup table for each frame index and for each feature vector component, and then sum up the appropriate values from the table. The same approach is equally applicable with the VQHMM technique.

With VQHMMs, it is also possible to speed up the observation probability computations by pre-calculating the probabilities for each class of densities at a time. The formula for the Mahalanobis distance calculation, that is the most expensive part in the state observation probability calculation, can be expressed in the form

$$\sum_{i=1}^N \frac{(x_i - \mu_i)^2}{\sigma_i^2} = \sum_{i=1}^N \left(\frac{x_i}{\sigma_i} - \frac{\mu_i}{\sigma_i} \right)^2. \quad (1)$$

It can be seen that by normalizing the feature vector and the mean codebook (normalized mean values can be stored in advance) with the given variance vector, the number of calculations needed is reduced approximately by one fourth if the number of mixtures is large.

4. Experimental evaluations

The baseline CDHMMs, qHMM, and the VQHMM acoustic model representations were tested on the isolated word recognition task. The front-end based on 12 static Mel Frequency Cepstral Coefficients (MFCC) coefficients and log-energy, together with their first- and second-order time derivatives was used. Feature vector normalization [8] was applied to the extracted features. Mean normalization was applied to all the components, but only the log-energy related parameters were variance normalized. In all cases, HMMs had three states with the left-to-right structure and with eight mixtures per state. The multilingual, baseline CDHMMs were trained on clean speech with speech data for all the languages.

The recognition tests were carried out both in clean and in noisy environments with and without speaker adaptation. For the noisy tests car noise, cafeteria, airport noise and music were mixed with different Signal-to-Noise Ratio (SNR) value to the original clean speech sample. The SNRs ranged between +20 and +5dB uniformly.

The qHMMs were obtained directly from the baseline CDHMMs by training two Lloyd-Max scalar quantizers and then quantizing the models. In the experiments, 6 bit representation for the mean components and 3 bit representation for the variance components was adopted. The variance vector quantization levels for the VVQ were set to 64. The number of variance vectors (hence the number of mixtures) for the baseline models was 1608.

Language	Clean, not adapted			Clean, adapted		
	Baseline	QHMM	VQHMM	Baseline	QHMM	VQHMM
Eng	95.2	95.0	95.2	98.5	98.5	98.6
Fre	94.9	95.0	95.3	97.8	97.9	98.1
Por	93.1	92.8	93.0	97.5	97.6	97.4
Spa	98.7	98.5	98.7	99.6	99.6	99.8
Average	95.5	95.3	95.5	98.4	98.4	98.5

Language	Noise, not adapted			Noise, adapted		
	Baseline	QHMM	VQHMM	Baseline	QHMM	VQHMM
Eng	90.9	90.6	90.1	95.9	95.8	95.7
Fre	90.8	90.7	91.0	95.8	96.1	96.1
Por	82.2	82.1	82.6	90.9	90.2	91.0
Spa	96.5	96.4	96.4	98.8	98.7	98.8
Average	90.1	90.0	90.0	95.4	95.2	95.4

Table 3: Comparison between the baseline, qHMM and VQHMMs ($m=6$, $v=3$ and $Q=64$).

Table 3 shows the performance of the models for all the test cases. It can be seen that there is no performance degradation with VQHMMs when comparing the recognition accuracy to the CDHMMs or qHMMs.

With the test settings used, the VQHMMs are of size 55.6 kB against the 72.2 and 248.1 kB for the qHMMs and the CDHMMs, respectively. Hence, the memory savings compared to the CDHMMs is 77.6% and compared to the qHMMs 23.0%. For the small devices with strict memory limitations the saving is essential.

5. Conclusions

In this paper, vector quantization, parameter tying and scalar quantization methods were combined in order to reduce the memory footprint and also the computational complexity of the acoustic HMMs. It was shown that the proposed method does not degrade the recognition performance compared to the CDHMM or qHMMs if quantization levels in case of vector and scalar quantization are properly chosen.

In the future the effect of different VQ algorithms for the method will be studied. It is also not needed to perform the vector quantization step for full N dimensional variance vectors but the vectors can be divided into parts. Hence also the variance vector splitting will be of interest in the future.

6. References

[1] V. V. Digalakis, P. Monaco, H. Murveit. "Genones: Generalized Mixture Tying in Continuous Hidden Markov Model-Based Speech Recognizers". In IEEE Transac. on Speech and Audio Processing, vol. 4, pp. 281-289, 1996

[2] J. Pan, B. Yuan, Y. Yan. "Effective Vector Quantization for a Highly Compact Acoustic Model

for LVCSR". In Proc. ICSLP, vol. 4, pp. 318-321, 2000.

[3] J. Kim, R. Haimi-Cohen, F. Soong. "Hidden Markov Models with Divergence Based Vector Quantized Variances". In Proc. of ICASSP, vol. 1, pp. 125-128, 1999

[4] J. Köhler. "Comparing three methods to create multilingual phone models for vocabulary independent speech recognition tasks" In Proc. of MIST Workshop, pp. 79-84, 1999.

[5] M. Vasilache. "Speech Recognition Using HMMs with Quantized Parameters". In Proc. of ICSLP, vol. 1, pp. 441-444, 2000.

[6] R. P. Lippmann, E. A. Martin, D. B. Paul, "Multi-style Training for Robust Isolated-word Speech Recognition", Proc. of ICASSP, pp. 692-695, 1987.

[7] K. Filali, L. Xiao, J. A. Bilmes. "Data-Driven Vector Clustering for Low-Memory Footprint ASR". In Proc. of ICSLP, pp. 1601-1604, Denver, 2002.

[8] O. Viikki, D. Bye, K. Laurila. "A Recursive Feature Vector Normalization Approach for Robust Speech Recognition in Noise". In Proc. of ICASSP, pp. 733-736, Seattle, 1999.