EXPLOITING ACOUSTIC FEATURE CORRELATIONS BY JOINT NEURAL VECTOR QUANTIZER DESIGN IN A DISCRETE HMM SYSTEM

Christoph Neukirchen, Daniel Willett, Stefan Eickeler, Stefan Müller

Department of Computer Science
Faculty of Electrical Engineering
Gerhard-Mercator-University Duisburg, Germany
e-mail: {chn,willett,eickeler,stm}@fb9-ti.uni-duisburg.de

ABSTRACT

In previous work about hybrid speech recognizers with discrete HMMs we have shown that VQs, that are trained according to an MMI criterion, are well suited for ML estimated Bayes classifiers. This is only valid for single VQ systems. In this paper we extend the theory to speech recognizers with multiple VQs. This leads to a joint training criterion for arbitrary multiple neural VQs that considers the inter VQ correlation during parameter estimation. The idea of a gradient based joint training method is derived. Experimental results indicate that inter VQ correlations can cause some degradation of recognition performance. The joint multiple VQ training decorrelates the quantizer labels and improves system performance. In addition the new training criterion allows for a less careful way of splitting up the feature vector into multiple streams that do not have to be statistically independent. In particular the usage of highly correlated features in conjunction with the novel training criterion in the experiments leads to important gains in recognition performance for the speaker independent Resource Management database and gives the lowest error rate of 5.0% we ever obtained in this framework.

1. INTRODUCTION

In previous work [5] and [3] we have presented a theoretical framework for a hybrid speech recognition system that consists of discrete HMMs and a general Winner-takes-all neural network that serves as VQ. The neural network maps the continuous feature vector $\mathbf{x}$ on a discrete label $\hat{m} = \hat{m}(\mathbf{x})$ that is used as a discrete feature in the subsequent discrete HMM system where $p(\hat{m}|w)$ is evaluated (w represents an HMM state). The idea in [5] is to train the HMMs and the neural network (i.e. the VQ) simultaneously by considering the combination of VQ and HMM as a single continuous classification system. Given a fixed Viterbi alignment of the training data by HMM states (or phonemes) it has been shown that maximum likelihood training of the combined system (i.e. $\operatorname{argmax}_\theta \sum_n \log p_\theta(\mathbf{x}(n)|w(n))$) leads to the maximum mutual information (MMI) criterion for the parameter estimation of the neural VQ. Thus the weights of the neural network must be set in order to maximize the objective function given by $I_\theta(\hat{M}, W) = H_\theta(\hat{M}) - H_\theta(\hat{M}|W)$. An algorithm that is similar to the conventional back-propagation method that achieves this parameter optimization by gradient descent is derived in [3].

In practice this hybrid system performs very well on the speaker independent Resource Management (RM) and Wall Street Journal (WSJ) continuous speech databases (see [3],[6]). It outperforms systems based on discrete HMMs and is comparable in recognition rates to continuous HMM systems, while being faster due to discrete local likelihood calculation.

Differing from the theory given above and in [5] (where one single VQ in the classification system is assumed) the hybrid speech recognizer makes use of four different VQs, each of them quantizing one kind of acoustic feature (e.g. cepstrum, Δ-cepstrum, etc.). For reasons of simplification, during speech system design each neural VQ was trained according to the MMI-criterion independently from the other VQs. Thus each single neural VQ is optimal for that system in the ML-sense, but the combination of the four VQs might be not, since the correlations between the different VQ labels are not taken into account.

In the following, the theoretical framework for hybrid systems based on neural VQs and discrete classifiers is extended to the integration of multiple VQs that allows for the usage of highly correlated acoustic features in speech recognition systems.

2. SINGLE VS. MULTIPLE VQ

In pattern recognition systems based on discrete models (e.g. HMMs with discrete output probabilities) local likelihood calculation is a two stage process: i) a vector quantizer, that partitions the continuous feature space into distinct regions, maps a given feature vector on a discrete label ($\mathbf{x} \rightarrow \hat{m}$). ii) that discrete label is used to obtain the local likelihood $P(\hat{m}|w)$ (where w is a pattern class or an HMM state in a speech recognition system, the number of different classes (or states) is denoted K).

The so called codebook size (denoted J), i.e. the number of regions the feature space is divided into by the VQ, is crucial for classifier performance. The larger the codebook size, the higher is the resolution of the VQ in the classification system. Thus, if the size of the codebook is chosen too small, many details of the feature space structure are neglected and classifier performance will be bad. On the other hand, the number of parameters to be estimated in the VQ and in the discrete classifier (what might become dominant in the case of many classes, e.g. triphone states) increases with the codebook size. Thus, the choice of the codebook size must be a compromise between feature space resolution and number of parameters that can be estimated properly.

In speech recognition the usage of multiple VQs in combination with discrete HMMs is quite common [1]. In this case the components of a given feature vector $\mathbf{x}$ are used to form Z different subvectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(Z)}$ that contain the original features of $\mathbf{x}$. E.g. $\mathbf{x}^{(1)}$ contains the cepstral features, $\mathbf{x}^{(2)}$ contains the Δ-cepstral features, etc. Each of these subvectors are mapped on a discrete VQ label by an individual VQ ($\mathbf{x}^{(z)} \rightarrow \hat{m}^{(z)}$). Thus, the feature vector $\mathbf{x}$ is mapped on a set of discrete labels $\hat{m}^{(1)}, \ldots, \hat{m}^{(Z)}$. If the codebook size of the z-th VQ is given by $J^{(z)}$, the original feature space is divided by the set of multiple VQs into $\prod_{z=1}^{Z} J^{(z)}$ separate regions. Hence, the usage of multiple VQs increases the resolution of the vector quantizing stage.

In the discrete pattern classifier the set of multiple VQ labels is used to obtain the local likelihood. To simplify calculations and to reduce the number of parameters in the classifier, in many cases the class dependent distribution of the different VQ labels are assumed statistically independent:

$$P(\hat{m}|w) = P(\hat{m}^{(1)}, \ldots, \hat{m}^{(Z)}|w) = \prod_{z=1}^{Z} P(\hat{m}^{(z)}|w) \quad (1)$$

Therefore the number of parameters (i.e. probabilities) per class in the discrete classifier is given by $\sum_{z=1}^{Z} J^{(z)}$ which is smaller than the number of VQ partitions in general. The class dependent probabilities assigned to each VQ partition cannot be chosen independently since they depend on this smaller number of parameters. If the set of VQ labels is actually statistically independent (e.g. if the $\mathbf{x}^{(z)}$, $1 \leq z \leq Z$ are independent) Eqn. (1) works fine. However, if the labels generated by the different VQs are correlated, Eqn. (1) assigns a wrong probability to the VQ partitions, that causes a degradation in recognition performance.

3. VQ PARAMETER ESTIMATION FOR CORRELATED FEATURES

3.1. Independent VQ training

As mentioned above, many speech recognition systems make use of multiple VQs. This increases the resolution of the vector quantizer without increasing the number of parameters to be estimated too much. The hybrid speech recognizer described in [6] makes use of four ($Z = 4$) different neural VQs: i) for 12 cepstral parameters, ii) for 12 Δ-ceps, iii) for 12 $\Delta\Delta$-ceps, iv) for pow.+Δ-pow+$\Delta\Delta$-pow . That way of splitting up the original acoustic feature vector is common and has been used in several different speech systems (e.g. see [1]) with success. This choice of feature splitting seems to be reasonable since these four subvectors are known to be quite uncorrelated. Hence, the four VQ labels $\hat{m}^{(1)}, \ldots, \hat{m}^{(4)}$ generated by the neural networks are (nearly) statistically independent for a given class (i.e. HMM state). Thus Eqn. (1) works quite well for that case.

During training of the systems in [6] and [3], the parameters of each of the four neural network VQs are estimated according to the MMI objective function independently from each other (i.e. $\mathrm{argmax}_\theta\, I_\theta(\hat{M}^{(z)}, W)\ \forall\ z$). However, independent neural VQ training is a simplification that is not covered by the framework presented in [5] where only one single VQ is considered. In that independent training case, the information theoretic objective function that was used for training the set of neural VQs can be written as:

$$\sum_{z=1}^{Z} I_\theta(\hat{M}^{(z)}, W) \quad (2)$$

In this objective function, no interaction between the different VQs is considered. Thus, each one of the VQs fits very well to the classification system on its own, but the behavior of the set of VQs is unclear. If the form of the subvectors $\mathbf{x}^{(z)}$ is chosen carefully such that they are statistically independent, interaction between different VQs should be unimportant. However, it is difficult to select totally uncorrelated features in advance.

In a different hybrid speech recognition system (see [2]), that uses MLPs to estimate local posteriors, the incorporation of several adjacent frames of acoustic feature vectors has given much improvement compared to the single frame features. Thus the acoustic information contained in a long span of adjacent frames seems to be important for speech recognition. Several adjacent frames can be simply integrated in a multiple VQ system by extending the input of the VQs to several frames of acoustic subvectors (e.g. for the z-th VQ: $\ldots, \mathbf{x}^{(z)}(t-1), \mathbf{x}^{(z)}(t), \mathbf{x}^{(z)}(t+1), \ldots$). If the different VQs process multiple frames of Δ- and $\Delta\Delta$-features etc., then the VQ inputs become more correlated (since the different subvectors can be derived from each other) and the VQ outputs may be not statistically independent. In this case Eqn. (1) fails and independent neural VQ parameter estimation is not optimal.

3.2. Single VQ training

One way to avoid problems with (possibly) correlated acoustic features, is to process all those that might be correlated in a single VQ. If adjacent frames of static and Δ-features etc. are incorporated, all these features are processed by one single VQ. In this case no independence assumptions (like in Eqn. (1) must be made. In addition, the MMI framework [5] for training the neural VQ parameters remains valid without modifications. But to achieve a VQ-resolution for that single VQ that is comparable to the multi VQ system, the codebook size must be chosen very large. This increases the number of parameters to be trained in the neural network and, what might be more important, in the classifier (i.e. discrete HMM). Thus robust parameter estimation with limited training data might become intractable.

3.3. Joint VQ training

According to the theory in [5] the parameters of the neural VQ must be estimated in order to maximize $I_\theta(\hat{M}, W)$ if the likelihood of the entire system consisting of the VQ and the discrete classifier has to be maximized. As given above in the case of multiple VQs, the single VQ label $\hat{m}$ is replaced by the set of VQ labels $\hat{m}^{(1)}, \ldots, \hat{m}^{(Z)}$. Hence, the objective function for training multiple neural VQs under consideration of feature correlations between different VQs becomes $I_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)}, W) = H_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)}) - H_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)}|W)$. If the discrete classifier makes use of the classwise independence assumption of the different VQ labels (i.e. Eqn. (1)) the class conditional entropy can be written as:

$$H_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)}|W) = \sum_{z=1}^{Z} H_\theta(\hat{M}^{(z)}|W) \quad (3)$$

In this case the objective function can be transformed into:

$$I_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)}, W) = -I_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)}) + \sum_{z=1}^{Z} I_\theta(\hat{M}^{(z)}, W) \quad (4)$$

This is a generalization of the expression found in other work (see [4]) where only the parameters of a single VQ are estimated. When comparing Eqn. (4) with Eqn. (2) it can be seen that independent VQ training is a simplification of the full criterion that neglects the correlations between the different VQ labels (i.e. $I_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)})$). The full multi VQ criterion (Eqn. (4)) tries to optimize each of the neural VQs according to the theoretical framework of [5] while minimizing the inter-VQ-label correlations simultaneously. Thus if the subvectors, used as multi VQ input, are uncorrelated the VQ labels are statistically independent and Eqn. (4) is equivalent to the separate independent VQ MMI-training (since $I_\theta(\hat{M}^{(1)}, \ldots, \hat{M}^{(Z)})$ equals zero). On the other hand in the case of joint VQ training, a very careful independent subvector choice is not necessary because during training the different VQ labels are decorrelated as far as possible.

For training the neural network VQs jointly, the criterion Eqn. (4) must be maximized. This can be done in a way similar to [3]

by calculating the derivative of Eqn. (4) with respect to any neural network parameter θ. Since the derivative of the right hand sum (i.e. optimization of different independent VQs) in Eqn. (4) has been already derived in [3], only $\frac{\partial}{\partial\theta}(-I_\theta(\hat{M}^{(1)},\ldots,\hat{M}^{(Z)}))$ needs to be introduced here. The mutual information between the different VQ labels can be written as:

$$I_\theta(\hat{M}^{(1)},\ldots,\hat{M}^{(Z)}) = \sum_{j_1=1}^{J_1}\cdots\sum_{j_Z=1}^{J_Z} P^*_\theta(\hat{m}^{(1)}_{j_1},\ldots,\hat{m}^{(Z)}_{j_Z})\cdot \cdot\log\frac{P_\theta(\hat{m}^{(1)}_{j_1},\ldots,\hat{m}^{(Z)}_{j_Z})}{\prod_{z=1}^{Z}P_\theta(\hat{m}^{(z)}_{j_z})} \quad (5)$$

Here $P^*_\theta(\hat{m}^{(1)}_{j_1},\ldots,\hat{m}^{(Z)}_{j_Z})$ is the true VQ label distribution in the training data, and $P_\theta(\hat{m}^{(1)}_{j_1},\ldots,\hat{m}^{(Z)}_{j_Z})$ is the probability seen by the Bayes classifier, i.e.:

$$P_\theta(\hat{m}^{(1)}_{j_1},\ldots,\hat{m}^{(Z)}_{j_Z}) = \sum_{k=1}^{K}P(w_k)\cdot\prod_{z=1}^{Z}P_\theta(\hat{m}^{(z)}_{j_z}|w_k) \quad (6)$$

Using the chain rule, the derivative of Eqn. (5) with respect to any neural VQ weight θ can be found in a way similar to that presented in [5]. To circumvent the problems with the non-continuous nature of the VQ-function, the Winner-takes-all output is approximated by a Softmax function (similar to [3]).

4. EXPERIMENTS AND RESULTS

4.1. Test conditions

To compare the effects of independent and joint VQ training and the different ways to divide the acoustic features, several neural network VQs are trained. Although any kind of Winner-take-all neural network can be used as VQ in this framework, for reasons of simplicity in all the tests described here we limit the choice to single layer perceptron neural networks, i.e. the VQ boundaries are always linear.

As acoustic features every 10 ms 12 MFCC coefficients and the signal power are extracted from the speech signal. As dynamic features, Δ- and $\Delta\Delta$-coefficients are generated comprising 39 features per frame.

For the training of the neural network VQs and the HMMs we use the 3990 speaker independent sentences of the Resource Management (RM) database. To compare the system performance recognition results are obtained for the official Feb'89, Oct'89, Feb'91 and Sep'92 DARPA RM speaker independent test sets. Recognition is done via a beam search guided Viterbi decoder using the DARPA word pair grammar (perplexity: ca. 60). Word error rates (WER) are given as average over these four test sets.

The HMM speech recognizer makes use of strictly left-to-right tree state HMMs. In all the tests monophone and triphone results are given. The monophone system consists of 49 different context independent HMMs. The triphone system consists of 2309 context dependent HMMs that are word internal only. To balance the number of HMM parameters against the amount of training data, the triphone states are tied via a phonetically based decision tree. By this method, in the following experiments the number of triphone states is always chosen in order to give maximum recognition performance. For all test conditions the number of parameters used in the single-layer perceptron VQs and for the HMMs are approximately given.

			monophones		wrd. int. triph	
# frms	cdb size	VQ parm	WER	HMM parm	WER	HMM parm
1	4·200	9k	13.6%	116k	6.1%	1.6M
3	4·200	24k	12.3%	116k	5.8%	1.6M
7	4·200	55k	11.7%	116k	5.8%	1.6M

Table 1. System with four different neural VQs that are trained independently. The number of adjacent input feature frames is varied.

4.2. Independent VQ training

The first series of experiments uses the traditional method of [3] for neural VQ parameter estimation, i.e. the neural networks are trained independently of each other according to the MMI criterion as given in Eqn. (2). The first neural VQ quantizes the cepstral features, the second one the Δ-features, the third one the $\Delta\Delta$-features and the fourth one all the energy related features. In previous experiments, a codebook size of 200 for the VQs has given the best results for the triphone system, while better monophone results can be achieved for larger codebook sizes. Hence for simplicity the codebook size is fixed to 200 here. The experimental results for a various number of adjacent VQ input frames are shown in Table 2. It can be seen that using 3 or 7 adjacent instead of single frame input features improves the monophone recognition rate, in spite of introducing higher input feature correlations due to incorporation of adjacent frames. For the triphone HMMs the input of 3 adjacent frames gives the best result of 5.8% word error rate, while using more frames does not improve performance. This may be due to the higher VQ output correlations, but the reason is not totally clear since the monophone results are not degraded. It must be noted that the number of HMM parameters is not increased by using a larger number of frames. In all these cases the number of VQ parameters remains quite small.

4.3. One single VQ for all features

In a second experimental setup all acoustic features are processed by one single neural VQ that is trained to maximize the MMI criterion. Thus, inter-VQ correlations cannot cause degradations. While in the previous experiments the number of partitions, the feature space is divided into by the VQ, was fixed to 200^4, in this experiment the optimal codebook size must be be determined. Table 2 shows the error rates for that system with single frame feature input and varying codebook sizes. The best single frame result for triphone HMMs (6.7%), is obtained with a codebook size of 1000. For this configuration the number of HMM parameters is very large (4 million). In this case the VQ resolution is coarse compared to the multi VQ system in the previous experiments. However, increasing the resolution (i.e. the codebook size) would also increase the number of HMM parameters. For a larger codebook size (1500) the triphone recognition rate drops while the monophone system (using much fewer parameters) still improves.

For a fixed codebook size of 1000 the single neural VQ system performs best with 5 adjacent frames of the acoustic feature vectors. In this case the monophone result (9.9%) is better than the best results obtained by the 4 independent VQ system in the previous experiments. However, all the triphone error rates are higher compared to the multi VQ system shown in Table 1. This may be due to the larger number of parameters and the poor VQ resolution.

4.4. Joint VQ training

In the final series of experiments four neural VQs are trained jointly with respect to their VQ label correlations according to Eqn. (4).

			monophones		wrd. int. triph	
# frms	cdb size	VQ parm	WER	HMM parm	WER	HMM parm
1	200	8k	14.1%	29k	8.3%	0.8M
1	500	20k	11.9%	73k	7.1%	2.0M
1	1000	40k	11.3%	145k	6.7%	4.0M
5	1000	196k	9.9%	145k	6.4%	4.0M
1	1500	60k	10.7%	218k	6.9%	6.0M

Table 2. System with one single neural VQ for all features. The codebook size and the number of adjacent input feature frames is varied.

			monophones		wrd. int. triph	
# frms	cdb size	VQ parm	WER	HMM parm	WER	HMM parm
3	4·200	24k	11.0%	116k	5.5%	1.6M
7	4·200	55k	10.6%	116k	5.5%	1.6M

Table 3. System with four different neural VQs that are trained jointly. Each VQ uses different input features. The number of adjacent input feature frames is varied.

Hence, if VQ correlations had caused performance degradations in the experiments of Table 1, these should be reduced now. To keep the number of HMM parameters comparable to those of the initial experiments, the codebook sizes are fixed to 200 for all VQs. Due to the very CPU-expensive joint VQ training method only a few experiments have been made.

4.4.1. Splitting up the feature vector

As in the first experiments, the features are divided into four different subvectors containing cepstral parameters, Δ-cep., $\Delta\Delta$-cep. and energy related features. To introduce correlations to the VQ input, 3 and 7 adjacent feature frames are considered. The results are given in Table 3. The comparison of the independent and the joint neural VQ training shows, that for monophones the word error rate drops from 12.3% to 11.0% (for 3 adjacent frames) and from 11.7% to 10.6% (for 7 frames). In both cases the total number of parameters is the same. Thus training the neural networks jointly, improves speech recognition performance. For the triphone system, a small reduction of the word error rates from 5.8% to 5.5% can be observed. Unfortunately, for the triphone system no improvement was gained again when increasing the number of adjacent feature frames from 3 to 7. This effect can also be observed for the independently trained VQs (see Table 1). Since the joint VQ training method should reduce the correlations between the VQs, this problem may be triphone specific and not correlation dependent only.

4.4.2. Multiple feature vector usage

In the initial experiments with independent VQ training the subvectors that are fed into the different VQs are chosen very carefully. In the case of joint VQ training the different VQ labels are much less correlated as forced by the training criterion. Hence, different VQs with highly correlated feature inputs may be used. As an extreme all the VQs may use the same input feature vector. In a final experiment, four different neural VQs with codebook size 200 are trained jointly with the same input features. In this case, the feature vector used in all VQs consists of 7 adjacent frames of 12 cepstral and energy parameters and their Δ and $\Delta\Delta$ components (273 components totally). Hence, the number of VQ parameters is quite large. Table 4 shows the recognition results. While using the same number of HMM parameters the monophone results are much better compared to the system in Table 3 (8.8% instead of 10.6%). The monophone error rate is also lower compared to the single neural VQ system (Table 2). Although the great improvements for the monophones cannot be directly transfered to the triphone system, the error rate for the context dependent system drops significantly to 5.0%. This is the lowest word error we ever obtained for the hybrid MMI-connectionist VQ / discrete HMM system on this task with word internal models.

			monophones		wrd. int. triph	
# frms	cdb size	VQ parm	WER	HMM parm	WER	HMM parm
7	4·200	219k	8.8%	116k	5.0%	1.6M

Table 4. System with four different neural VQs that are trained jointly. Each VQ uses the same input vector.

5. CONCLUSIONS

The correlations between VQ labels can be the source of degradation of recognition performance. As shown by theory and the experimental results, the joint training of multiple neural MMI-VQs or the incorporation of one single neural VQ for all features can be used to cope with these correlations. The experiments suggest that multi VQs seems to provide a better compromise between VQ resolution and the total number of parameters to estimate compared to the single VQ system. For the triphone system mutltiple VQs outperform a single VQ. Joint neural VQ training can be embedded in the traditional gradient based MMI training framework of arbitrary Winner-take-all neural networks presented in [3].

An additional advantage of training multiple neural VQs jointly comes from the evidence that the choice of decorrelated multi VQ input features seems to be quite uncritical. In our case the best recognition result is obtained when using four different (jointly trained) VQs that have the same input feature vector. Thus, the inputs are totally correlated. For this system the error rate is even lower than for the one presented in [3] that makes use of multilayer neural networks as VQ (hence VQ boundaries are non-linear).

6. ACKNOWLEDGMENTS

This work was partly supported by the DFG (German Research Foundation) under contract Ri 658/6-1. Responsibility for the content of this paper is with the authors.

REFERENCES

[1] K.F. Lee, et al, "An Overview of the SPHINX Speech Recognition system", *IEEE Trans. Acoust, Speech, Signal Processing*,Vol. 38, No. 1, Jan. 1990, pp. 35–45.

[2] N. Morgan, H. Bourlard, "Neural Networks for Statistical Recognition of Continuous Speech," *Proc. IEEE*, Vol. 83, No. 5, May 1995, pp. 742–770.

[3] Ch. Neukirchen, G. Rigoll, "Advanced training methods and new network topologies for hybrid MMI-Connectionist/HMM speech recognition systems", *Proc. IEEE-ICASSP*, 1997, pp. 3257–3260.

[4] M. Osterndorf, J.R. Rohlicek, "Joint quantizer design and parameter estimation for discrete Hidden Markov Models", *Proc. IEEE-ICASSP*, 1990, pp. 705–708.

[5] G. Rigoll, Ch. Neukirchen, "A new approach to hybrid HMM/ANN speech recognition using mutual information neural networks", *Advances in Neural Information Processing Systems 9*, NIPS*96, Denver, 1996, pp. 772–778.

[6] J. Rottland, et al, "Large Vocabulary Speech Recognition with Context dependent MMI-Connectionist/HMM Systems using the WSJ Database", *Proc. Eurospeech*, 1997, pp. 79–82.