

Low-dimensional Space Transforms of Posteriors in Speech Recognition

Jan Zelinka, Jan Trmal, Luděk Müller

The Department of Cybernetics, University of West Bohemia, Czech Republic

zelinka@kky.zcu.cz, jtrmal@kky.zcu.cz, muller@kky.zcu.cz

Abstract

In this paper we present three novel posterior transforms with the primary goal to achieve a high reduction of a feature vector size. The presented methods transform the posteriors to 1 D or 2 D space. For such a high reduction ratio the usually applied methods fail to keep the discriminative information. Contrary, the presented methods were specifically designed to retain most of the discriminative information. In our experiments, we used several different combinations of feature extraction methods nowadays commonly used, i.e. the PLP features (augmented with delta and acceleration coefficients) and two kinds of MLP-ANN features: the bottleneck (BN) and posterior estimates (PE). The experiments were designed with special attention to the assessment of possible improvements of the performance when the PLP features are combined either with the BN features or with the PE features whose dimensionality was reduced using the proposed feature transforms. The performance of the designed transforms was tested on two different speech corpora: a telephone speech SpeechDat-East corpus and multimodal Czech Audio-Visual corpus.

Index Terms: speech recognition, posteriors, ANN, bottleneck

1. Introduction

There are two main kinds of ANN application to speech recognition. The first one is an HMM hybrid where posterior probabilities¹ estimates are used instead of the observation probability density produced by GMM. In the second approach, the MLP-ANN serves as a front-end producing features for GMM/HMM based speech recognition system. This paper is focused only on this second approach.

The features can be obtained in several different ways. We used two different approaches. In the first approach, the feature vectors are generated directly from the posteriors. Using the posteriors as a feature vector is not practical because of the high dimensionality of the posteriors.

Usually, the common dimension reduction methods (PCA, LDA, HLDA, etc.) are used for this task. Moreover, before the dimension reduction the individual elements of the posteriors are preprocessed separately by a non-linear function. The reasoning of the preprocessing is that it helps to improve the discriminative power of the features. Again, there is not a universally accepted choice of the preprocessing function. Ordinarily, the log function is used, however there are other useful transforms. For example, one could choose the inverse sigmoidal function or the function

$$f(x) = \frac{1}{x + \delta}, \quad (1)$$

¹In this paper posteriors stands for conditional probabilities $p(\omega|x)$ of some speech unit ω (such as phoneme or state of a monophone) given the occurrence of some observation feature vector x .

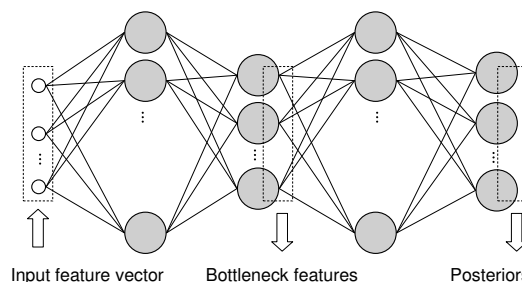


Figure 1: General scheme of BN features extraction.

where $\delta > 0$ is close to zero. Unfortunately, the common transforms (i.e. PCA, LDA, HLDA, etc.) do not produce beneficial low-dimensional features, especially when the original number of features (i.e. posteriors) is more than a few tens. Therefore, we devised the new, low-dimensional transforms.

The second approach deals with the direct production of low-dimensional features. For this approach, the MLP-ANN has a special topology. It has usually 4 or more layers and the central layer has (very) low dimension (see Fig 1). The ANN itself is trained either to approximate the identity function, i.e. $f(x) = x$ or to approximate the real posteriors. Resultant features are the outputs of the central, low dimensional, layer. These features are called *bottleneck features* [1]. The important fact is that the features produced by the network estimating the real posteriors produces discriminative features, i.e. features that keep most of the information needed to discriminate between the individual classes. This is fundamentally opposed to the approximation of the identity function approach that produces features retaining most of the information needed for reconstruction of the input vector.

The posteriori probability estimates play a significant role in multimodal speech recognition. In this particular field, each modality source is processed independently to produce the corresponding (representative) feature vector and consequently, these feature vectors are combined together. The process of combining all the features together is called *modalities fusion*. The choice of an appropriate fusion method is not trivial. Usually, it involves extensive experiments to evaluate the proper choice. Using posteriors as the features simplifies this task. In our audio-visual speech recognition system the modality fusion combines the posteriors (see Fig 3). In our experiments, we applied a fusion method aggregating several particular combinations of elementary fusion methods (such as entropy based fusion or geometric fusion).

As has already been mentioned, we used two different corpora – telephone speech corpus SpeechDat-East (SD-E) [2] and

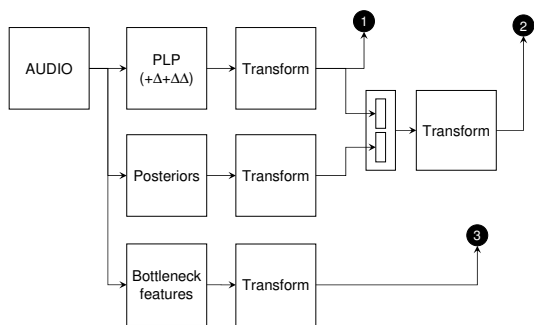


Figure 2: The scheme of feature vector for ASR generating – Audio speech recognition.

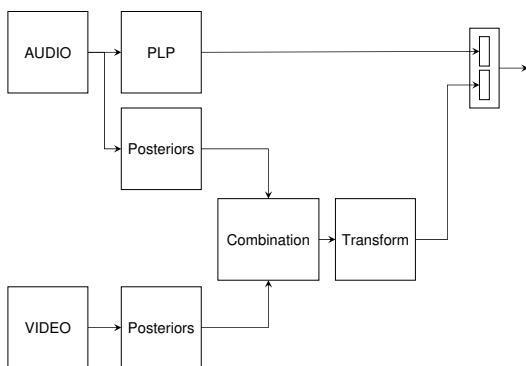


Figure 3: The scheme of feature vector for ASR generating – Audio-visual speech recognition.

the multimodal the Czech Audio-Visual Corpus (CAVC) [3]. Fig 2 shows six possible types of the final feature vector combinations. The transform for PLP (including Δ and $\Delta\Delta$ features) and BN features is either no transform (i.e. $f(x) = x$) or HLDA.

2. Posterior Estimates

ANN based approach to posterior estimation is definitely the most popular one although there are other ways such as the SVM based approach[4]. This paper discusses only the ANN based approach. The TRAPS [5, 6] method was used as an acoustic signal parameterization method for ANN based posterior estimate.

In this paper the problem of choice of the ANN structure is not discussed. Our ANN has a structure shown in Fig 1. The ANN for posterior estimation has two layers. The ANN for the BN features generation has four layers; the BN features are produced as the outputs of the second layer.

Sigmoid functions are chosen as the activation functions of neurons in all hidden layers. Activation functions in the output layer are softmax functions:

$$y_i = \frac{e^{\xi_i}}{\sum_j e^{\xi_j}}, \quad (2)$$

where y_i is the output of the i -th output neuron and ξ_i is the input of the i -th activation function. The number of features in input feature vector produced by the TRAPS signal parameterization method was 330. The number of neurons in the hidden layers was changed from 100 to 2500. The number of estimated posteriors was 55 or 111. ANN for BN features generation differs only in the ANN structure. Naturally, BN features are not computed by means of softmax function.

The error function used for ANN training was cross-entropy. Backpropagation or its variant is a common method of ANN training. Modifications often lie in dynamical change of the learning rate (such as the Bold Driver method) or in application of the momentum in the parameter update step. Our experiments lead us to conclusion that these methods usually have a poor convergence rate especially when an ANN with many parameters (1M and more) is trained. The backpropagation algorithm is a special case of gradient algorithm and therefore it can be interpreted as local criterion function approximation which is approximated by a linear hyper-plane. Nonetheless, there are optimization methods that approximate the criterion function using a quadratic hyper-plane. This approximation allows to find the location of an extreme faster and above all with no need for use a learning rate. Pure Newton's method or a quasi-newton method approximates the criterion function by a quadratic hyper-plane [7]. Certainly, the location of extreme of the criterion approximation is not the location of extreme of the criterion. Thus these methods are iterative. However the pure Newton's method cannot be applied for technical reason tied to the enormous size of the hessian matrix ($1M \times 1M$). Hence, approximation of the hessian using only the gradient – i.e. quasi-newton method – is suitable. The L-BFGS (Limited memory Broyden-Fletcher-Goldfarb-Shanno [8]) method² and iRPROP⁺ [9] method were used. Both these methods were significantly faster than the backpropagation with dynamic learning constant that we had used before. The ANN for BN features was trained using the iRPROP⁺ because the number of layers hidden layers with sigmoidal activation functions leads to gradients very close to zero and thus the L-BFGS algorithm failed to converge.

All the used ANN training algorithms are stochastic because of their random initialization, although all the steps s of all used ANN training algorithms are deterministic. The deterministic step s is given by the formula:

$$\theta_{t+1} = s(\theta_t), \quad (3)$$

where θ_t is a vector of ANN parameters. We designed and applied also a stochastic version of the deterministic step

$$\theta_{t+1} = s(\theta_t) + \gamma_t e_t, \quad (4)$$

where e_t is a random vector (with standard normal distribution) and γ_t is given by the formula

$$\gamma_t = \begin{cases} \gamma_0 & t = 1 \\ \gamma_{\uparrow} \gamma_{t-1} & \varepsilon(\theta_{t-1}) = \varepsilon(\theta_{t-2}) \\ \gamma_{\downarrow} \gamma_{t-1} & \varepsilon(\theta_{t-1}) \neq \varepsilon(\theta_{t-2}) \end{cases}, \quad (5)$$

where $\gamma_{\uparrow} > 1$ and $0 < \gamma_{\downarrow} < 1$ are the a priori chosen constants and ε is the monitored error function, which is in general not identical to the minimized error function. We designed this stochastic version as a compensation of the inferior initialization (i.e. θ_0).

²L-BFGS toolbox we used can be found at <http://www.chokkan.org/software/liblbfgs/>

3. Posterior Transforms

Standard posterior transforms, i.e. logarithm and linear transform such as PCA or HLDA, lead to a relatively high feature space dimension. On the other hand, the transforms we designed not only lead to a decrease of the word error rate of the speech recognition system (when the transformed posteriors are joined together with a usual feature vector) but they also lead to a very small feature vector dimension. The transforms are linear, they are applied as described by the following formula

$$y = M \cdot \mu(x), \quad (6)$$

where x is a vector of posteriors, M is the transformation matrix and μ is the function

$$\mu([x_1, \dots, x_{n_x}]^T) = [\delta_{x_1, x_{max}}, \dots, \delta_{x_{n_x}, x_{max}}]^T, \quad (7)$$

where $x_{max} = \arg \max_{i=1, \dots, n_x} x_i$ and $\delta_{a,b} = 1$ if $a = b$ and $\delta_{a,b} = 0$ otherwise.

It is easy to prove that there is a bijective linear function from a finite set $A \subset \mathfrak{R}^{n_x}$ to a set $B \subset \mathfrak{R}^{n_y}$ if set A and B have the same cardinality and if all elements of A are linearly independent. Evidently the elements of the set

$$\{\mu(x) | x \in \mathfrak{R}^{n_x}\} = \left\{ [\delta_{i,1}, \dots, \delta_{i,n_x}]^T | i = 1, \dots, n_x \right\} \quad (8)$$

are linearly independent. Distance base criteria for choosing the set B can be selected freely if there is no relevant addition information about classification given by (7). One of the possible criteria is called Discriminative Ratio (DR)

$$DR(B) = \frac{d(B)}{D(B)}, \quad (9)$$

where

$$d(B) = \min_{b_1, b_2 \in B, b_1 \neq b_2} \|b_1 - b_2\|, \quad (10)$$

$$D(B) = \max_{b_1, b_2 \in B, b_1 \neq b_2} \|b_1 - b_2\|. \quad (11)$$

The optimal set B maximizes the value of the function DR .

Firstly, we want to find B for $n_y = 1$. If $B \subset \mathfrak{R}$ and $d(B) = d$ then $D(B) \geq (n_x - 1)d$ and $DR(b) \leq \frac{1}{n_x - 1}$. Therefore the optimal set for $B \subset \mathfrak{R}$ is a set

$$B_a = \{a \cdot i + b | i = 0, \dots, n_x - 1\}, \quad (12)$$

where $a \neq 0, b \in \mathfrak{R}$. Consequently $DR(B_a) = \frac{1}{n_x - 1}$. Naturally we fixed $a = \frac{1}{n_x - 1}$ and $b = 0$. Secondly, we want to find B for $n_y = 2$. Unfortunately, there is no optimal solution for $B \in \mathfrak{R}^2$ for each $n_x \in N$. There are only some optimal solutions for only some particular $n_x \in N$. We proposed two suboptimal solutions B_a and B_c . The first set $B \subset \mathfrak{R}^2$ is the set

$$B_b = \left\{ \left[\sin\left(\frac{2\pi i}{n_x}\right); \cos\left(\frac{2\pi i}{n_x}\right) \right]^T | i = 0, \dots, n_x - 1 \right\}. \quad (13)$$

This set leads to

$$\sqrt{\frac{1 - \cos\left(\frac{2\pi}{n_x}\right)}{\frac{3}{2}}} \geq DR(B_b) \geq \sqrt{\frac{1 - \cos\left(\frac{2\pi}{n_x}\right)}{2}}. \quad (14)$$

The second solution is the set B_c of roughly square grid of points. In this case, $DR(B_c) = \frac{1}{\sqrt{2n_x}}$, where $\bar{n} \in N$ denotes the minimal square number which is greater than or equal

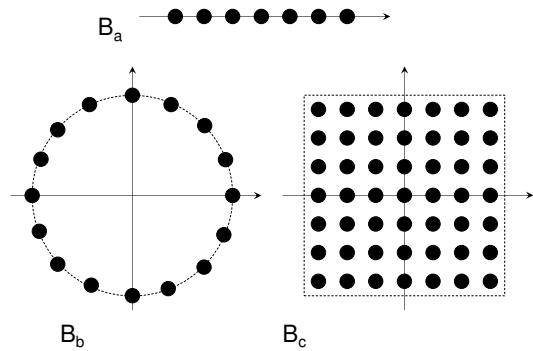


Figure 4: Illustration of the three designed transforms.

to $a \in N$ (e.g. $\overline{111} = 121$). Fig 4 shows all three proposed sets B .

Even though the sets A and B are fixed, the desired projection $A \mapsto B$ is not unambiguous but there are $n_x!$ different injective functions³, thus all possible projections cannot be investigated. We have decided to choose the projection where the sequence of points is given by the lexicographical order of units. (This does not probably minimize the main error function, i.e. word error rate, since it is possible that there are other projections such as a choice where the classes confused more often are geometrically closer than the other ones.) The used transformation matrices are:

$$M_a = \frac{1}{n_x - 1} \begin{bmatrix} 0 & \dots & n_x - 1 \end{bmatrix}, \quad (15)$$

$$M_b = \begin{bmatrix} \sin\left(\frac{2\pi \cdot 0}{n_x}\right) & \dots & \sin\left(\frac{2\pi(n_x-1)}{n_x}\right) \\ \cos\left(\frac{2\pi \cdot 0}{n_x}\right) & \dots & \cos\left(\frac{2\pi(n_x-1)}{n_x}\right) \end{bmatrix}, \quad (16)$$

$$M_c = \frac{1}{n} \begin{bmatrix} 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & \dots \\ 0 & 1 & \dots & n & 0 & 1 & \dots & n & \dots \end{bmatrix}, \quad (17)$$

where $n = \sqrt{\bar{n}_x} - 1$.

4. Experiments and Results

In our experiments two different corpora were used in our tests.

4.1. SpeechDat-East

In our experiments, the Czech part of SD-E was employed. The only modality is telephone speech recorded with 8 kHz sampling frequency. 700 speakers with 50 utterances for each speaker were reserved for training acoustic HMM. 150 speakers with 50 utterances for each speaker were reserved for testing. From each recording we removed a large portion of silence. This was done by forced-alignment of each utterance and the silences at the end and at the beginning of each utterance were removed. Speech units for which the posteriors were computed were the states of a monophone acoustic HMM. The number of posteriors was 111. The number of neurons in the only one hidden layer was 2500. The vocabulary for speech recognition consisted of 7737 words. No OOV word was present. We used 3-state triphone HMM with 8 GMM in each state. For

³ $111! \cong 1.76 \cdot 10^{180}$

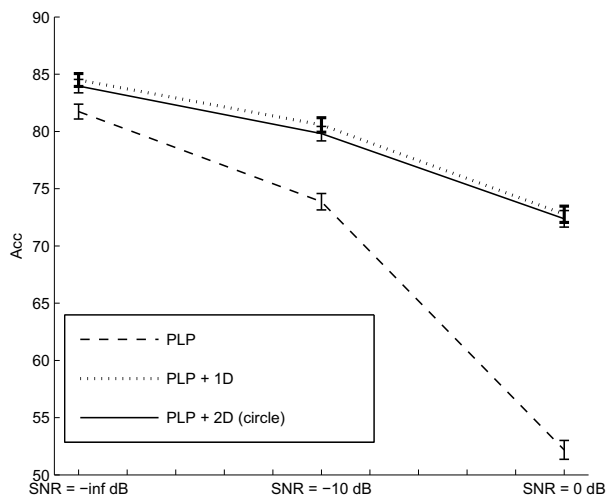


Figure 5: Contributions of proposed posterior transforms (CAVC).

the recognition a zerogram language model was applied. The results are shown in Table 1. The symbol + in the table denotes concatenation of feature vectors.

Table 1: Contributions of proposed posterior transforms (SDE).

Parameterization	Dimension	Accuracy
PLP	36	76.67%
HLDA(PLP)	36	78.61%
BottleNeck	30	74.19%
HLDA(BottleNeck)	30	75.51%
PLP + 1D	37	79.47%
PLP + 2D(circle)	38	79.48%
PLP + 2D(square)	38	79.36%
HLDA(PLP) + 1D	37	81.00%
HLDA(PLP) + 2D(circle)	38	80.87%
HLDA(PLP) + 2D(square)	38	81.07%
HLDA(PLP + 2D(square))	38	81.30%

4.2. Czech Audio-visual corpus

In CAVC the audio was recorded with 44.1 kHz sampling frequency. 150 sentences were reserved for training and 50 sentences were reserved for testing. Speech units for which the posteriors were computed were monophones. The number of posteriors was 55. The number of neurons in the only one hidden layer was 100. The vocabulary for speech recognition consisted of 345 words. No OOV word was present. Triphones were modeled using 3 states HMM with 8 GMM in each state. For the recognition a zerogram language model was applied.

Acoustic and video modalities were used in this experiment because the used posterior estimates are result of a combination of the posterior estimates computed from both modalities. Due to a small number of sentences in the test set we estimated 95% confidence intervals. The results are shown in Fig 5. We used Bootstrapping method to estimate the confidence intervals [10]. The confidence intervals are shown in Fig 5 as the error bars.

5. Conclusions

The results in Table 1 prove that concatenating the feature vector with transformed posteriors noticeably increases accuracy especially when no HLDA transformation is applied. Fig 5 shows that the used concatenations of transformed posteriors and PLP increase accuracy convincingly. Accuracy increments are considerable especially for $SNR > -\infty$. The occurrence of the increments is the reason to the audio-visual speech recognition is meaningful. Hence the designed posterior estimates and transforms and our experiments demonstrated that all three posteriors transforms are beneficial to speech recognition accuracy. But we are not able to positively decide which transform leads to the higher accuracy because the differences are not significant. Moreover, our experiments proved that L-BFGS quasi-newton method is practical for ANN based posterior estimate. Unfortunately, the performance of our BN features does not exceed the performance of PLP features. We explain this by the fact that the ANN generating the BN features is not adapted on the utterances with removed silence.

6. Acknowledgements

This research was supported by the Ministry of Education of the Czech Republic, project No. MŠMT LC536, by the Grant Agency of the Czech Republic, project No. GAČR 102/08/0707 and by the grant of The University of West Bohemia, project No. SGS-2010-054. The access to the META-Centrum supercomputing facilities provided under the research intent MSM6383917201 is highly acknowledged.

7. References

- [1] F. Grézil, M. Karafiát, and L. Burget, "Investigation into bottleneck features for meeting speech recognition," in *Proc. Interspeech 2009*, no. 9. International Speech Communication Association, 2009, pp. 2947–2950.
- [2] P. Pollak, "SpeechDat(E) – eastern european telephone speech databases," in *Proceedings LREC'2000 Satellite workshop XLDB*, Athens, Greece, 2000, pp. 20–25.
- [3] P. Císař, M. Železný, Z. Krňoul, J. Kanis, J. Zelinka, and L. Müller, "Design and recording of czech speech corpus for audio-visual continuous speech recognition," in *Proceedings of the Auditory-Visual Speech Processing International Conference 2005*. Vancouver Island: AVSP2005, 2005.
- [4] R. Solera-Urena, J. Padrell-Sendra, D. Martín-Iglesias, A. Gallardo-Antolín, C. Peláez-Moreno, and F. Díaz-De-María, "SVMs for automatic speech recognition: a survey," pp. 190–216, 2007.
- [5] P. Schwarz, P. Matějka, and J. Černocký, "Towards lower error rates in phoneme recognition," *Lecture Notes in Computer Science*, no. 3206, pp. 465–472, 2004.
- [6] —, "Recognition of phoneme strings using TRAP technique," *European Speech Communication*, vol. 2003, no. 9, pp. 1–4, 2003.
- [7] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. New York: John Wiley & Sons, 1987, ch. 8.7 : Polynomial time algorithms, pp. 183–188.
- [8] J. Nocedal, "Updating quasi-Newton matrices with limited storage," *Mathematics of computation*, vol. 35, no. 151, pp. 773–782, Jul. 1980.
- [9] I. C. and H. M., "Empirical evaluation of the improved Rprop learning algorithms," *Neurocomputing*, vol. 50, pp. 105–123(19), January 2003.
- [10] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection." Morgan Kaufmann, 1995, pp. 1137–1143.