Hierarchical Bottle Neck Features for LVCSR

Christian Plahl, Ralf Schlüter, Hermann Ney

Lehrstuhl für Informatik 6 - Computer Science Department
RWTH Aachen University, Aachen, Germany
{plahl,schlueter,ney}@cs.rwth-aachen.de

Abstract

This paper investigates the combination of different neural network topologies for probabilistic feature extraction. On one hand, a five-layer neural network used in bottle neck feature extraction allows to obtain arbitrary feature size without dimensionality reduction by transform, independently of the training targets. On the other hand, a hierarchical processing technique is effective and robust over several conditions. Even though the hierarchical and bottle neck processing performs equally well, the combination of both topologies improves the system by 5% relative. Furthermore, the MFCC baseline system is improved by up to 20% relative. This behaviour could be confirmed on two different tasks. In addition, we analyse the influence of multi-resolution RASTA filtering and long-term spectral features as input for the neural network feature extraction.

Index Terms: probabilistic features, bottle neck, hierarchical processing, LVCSR

1. Introduction

Phoneme posterior estimates derived from a neural network (NN) have recently become a major component of state-of-the-art automatic speech recognition (ASR) systems [1, 2]. Due to their different nature, they exhibit a large amount of complementary information. The role of the probabilistic features in ASR is thus to augment the cepstral features [1, 2, 3, 4]. Typically, NN based features are obtained by projecting a larger time span of a critical band spectrogram onto posterior probabilities of phoneme classes. In [5] several temporal trajectories of the critical bands are extracted and in [6] they are extented to obtain the RASTA processing. Nevertheless, short-term features like PLPs have been used as input features as well [7]. In order to better fit the subsequent Gaussian mixture model, the neural network estimates of posteriors are logarithmised and decorrelated by Principal Components Analysis (PCA) or Linear Discriminant Analysis (LDA), which also allow to reduce their dimensionality.

The structure of the neural networks has been under investigation as well. While in [8] a hierarchical processing of NNs is introduced to improve the final features, [9] proposes a bottle neck topology, where the posterior features are taken from an intermediate layer of the NN.

This paper investigates two different types of input features for neural network training and compares the bottle neck topology and the hierarchical processing. Furthermore, the combination of the two topologies is suggested, to benefit from the advantages of both approaches.

The paper is organised as follow: In Section 2 the two input feature types based on temporal patterns (TRAPs) and Multi-resolution RASTA (MRASTA) filtering are described. Next, the topology of the NN is presented in Section 3, followed by the acoustic modelling of the ASR systems in Section 4. The training and testing data are listed in Section 5. The paper ends with the experimental results in Section 6 and the summary and the conclusions in Section 7.

2. Neural Network Input Features

Most conventional ASR systems use short-term features such as MFCC or PLP features as input features [1, 2]. In contrast, long-term features, where the temporal context is up to one second, are used as additional features only, or as input features for an intermediate NN. Since the performance of probabilistic features is often below that of the standard cepstral features, probabilistic features are mostly augmented with the cepstral features of a baseline system. Furthermore, NN based features could improve the performance of LVCSR systems when used in combination with classical spectral features, as described in [4]. As shown in [3, 10] nowadays, probabilistic features achieve the same performance or can, in some cases, outperform the classical features. Nevertheless, results of systems using NN based features only are not considered.

2.1. TRAP-DCT Features

TempoRAl Patterns (TRAPs), as described in [5], are based on a huge temporal context. TRAP-based probabilistic features are formed by temporal trajectories of energies in independent critical bands. Since their introduction, several modifications targeting the input spectrogram have been proposed, [6, 11]. Moreover, the temporal context up to one second is often used.

In detail, the feature extraction is done as follows: Initially, coefficients of a short-term mel-scaled log-energy spectrogram are taken. Afterwards, mean and variance normalisation is applied to these features and the feature vector contains 19 coefficients. In a second step, a window of 51 frames (500ms) of these features are used to extract long energy trajectories for each of the 19 frequency subbands of the spectrogram. Next, these features are projected by a Discrete Cosine Transformation (DCT) and the first 16 coefficient including DC component are retained. Finally, the TRAP-DCT raw features contain $19 \times 16 = 304$ elements. These features are used as input to train the neural network posterior estimates.

2.2. Multi-resolution RASTA Features

Multi-resolution RASTA (MRASTA) filtering is an extension of the RASTA filtering introduced in [6]. This is achieved by applying two dimensional band-pass filters. Separate ranges of modulation frequencies are used to extract a set of multiple resolution filters. The RASTA filtering itself has been proposed as a modification of the TRAP-based probabilistic features extraction, introduced in [5].

At first, 19 critical bands are used from the critical band auditory spectrum, extracted from short-time Fourier transform of a signal every 10 ms. In a next step, each critical band is filtered with a bank of several low-pass filters represented by six first derivatives and six second derivatives of Gaussian functions. These Gaussian functions varying with variance in the range of 8-130 ms.

The MRASTA filtering subdivides the available modulation frequency range into separate channels with decreasing frequency resolution moving from slow to fast modulation. A detailed description of the Gaussian functions and of the NN features itself can be found in [12].

At the end, the features are used as input for a neural network or divided into the slow and fast modulation frequencies are used as input for the sequential and hierarchical processing, as described in Section 3.

3. Neural Network Topology

In a multi-class problem, NNs can be trained so that the output approximates class posterior probabilities. Generally, a 3-layer NN structure is used but other topologies have been investigated as well [8, 10].

Three different topologies of the neural network has been used for the experiments. As introduced in [8], the hierarchical processing of neural networks consist of several NNs where each network uses previously trained NN based posteriors as input features. Another concept estimates the NN based features from a intermediate layer instead of the final layer. This concept makes the size of the final features independent of the NN training phoneme targets [10]. The last concept combines the above two concepts to benefit from both advantages of the single concepts.

3.1. Single Hidden Layers

In this section we briefly describe neural networks with one single hidden layer. A 3-layer neural network consist of an input layer and a hidden layer and an output layer. The hidden layer is large to provide the necessary model power to reduce the final classification error [5, 7]. Moreover, in ASR the targets of the output layer are represented by phonetic units. As a results of the training of the neural network on phonetic targets posterior probabilities for each target could be derived. In order to better fit the subsequent Gaussian mixture model, the neural network estimates of posteriors are logarithmised and decorrelated by PCA or LDA. Next, a dimensionality reduction accounting for 95% of the total variability is applied.

3.2. Hierarchical Processing

A hierarchical processing of NNs is a cascade of NNs, where the next neural network uses the NN based posteriors from the previous NN as input features. Moreover, other features like the raw features used before could also be provided. As shown in [8] such a hierarchical processing improves the accuracy of the final posterior estimates and of the complete system trained with the improved posteriors. Figure 1 illustrates the hierarchical processing based on a 3-layer neural network.

In [12] the MRASTA features are split up into the fast and slow modulation frequencies. Moreover, the hierarchy used consists of two NN where the first net is trained on the fast modulation frequencies. The second NN uses the LOG/PCA transformed posterior estimates and the slow modulation frequencies as input.

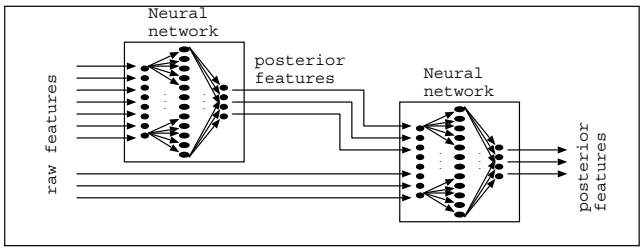

Figure 1: Hierarchical processing of neural network. Here the neural network consist of 3-layers.

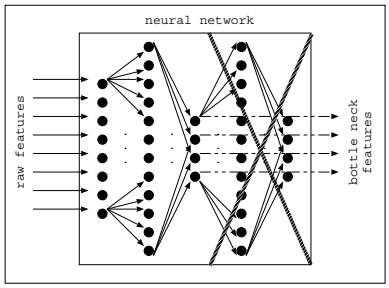

Figure 2: Bottle neck topology of a neural network. A full 5-layer network is trained, but the linear output of layer three (bootle neck) is taken only.

In our experiments we used the same concept. At first, a 3-layer neural network is trained using the fast modulation frequencies and the 19 critical band energies as raw features. Overall, the input vector contain of 235 elements. The second 3-layer net trained consists of the LOG/PCA transformed posterior features and the critical band energies and the slow modulation frequencies. Nevertheless, the transformed posterior estimates are fed in a sliding window of length nine to increase the contextual information. Finally, the posterior features are transformed by logarithm and PCA again. After PCA, a dimensionality reduction accounting for 95% of the total variability is applied. Overall, these features ends up with a dimensionality of 30 and are referred to as *HIER.MRASTA*.

3.3. Multi Hidden Layers - bottle neck

The concept of bottle neck feature has been first introduced in [9]. On the one hand, the goal has been to provide the ability to compress the input raw features in an abitrary size and on the other hand to ensure a good class separability of the output features.

In our experiments a 5-layer NN has been set up as illustrated in Figure 2. Since the first and last layer are needed for I/O interface, there are three hidden layers in the NN. The first hidden layer is large to provide the necessary modelling power. The size of the middle layer, the bottle neck, has been chosen equal to the number of nodes of the output layer. This allows a direct comparison between the final posterior features and the probabilistic features obtained from the bottle neck. The last layer has been again enlarged to further improve the classification error.

In order to analyse the behaviour of the bottle neck we used the TRAP-DCT and MRASTA features as input features described in Section 2. In the first experiment the MRASTA features have not been split up into the fast and slow modulation frequencies but augmented with the critical band energies. Af-

ter training the NN, the linear output of the bottle neck layer is normalized by mean and variance. Furthermore, these features are reduced by a PCA from 44 to 34 elements.

In the second experiment the MRASTA features are replaced by the TRAP-DCT features. The final probabilistic features are normalised and also reduced to 34 elements. The corresponding systems are labelled as *BNECK.TRAP-DCT* and *BNECK.MRASTA*. Experimental results are shown in Table 1 row three and four.

3.4. Hierarchical Bottle Neck Processing

The concept of hierarchical bottle neck features combines the advantages of the hierarchical and the bottle neck approach. Instead of using the 3-layer NN in the hierarchical processing we exchanged the NN with the bottle neck concept.

After training the first NN with the fast modulation frequencies, the second NN takes the bottle neck features and the slow modulation frequencies as input. Next, the linear output of the bottle neck layer of the second NN is normalized by mean and variance normalisation. These features are referred to as *HIER-BNECK.MRASTA* in the experimental section. In order to compare the results, the dimension of the HIER-BNECK.MRASTA is fixed to 34 elements.

4. Acoustic Modelling

In order to evaluate the relevance of the structure and the relevance of the input features of a neural network several full ASR systems based on Gaussian hidden Markov models (HMM) have been trained. Moreover, the systems differ only in the structure of the topology of the neural network or in the type of the input features used for training the neural network.

The systems built are based on the English system used in the QUAERO 2009 Evaluation, which is described in detail in [13]. The complex English system has been simplified. More precisely, the American Broadcast News (BN) part has been taken only. As shown in [13] the described systems are competitive to other state-of-the-art systems and has been trained with the RWTH Speech Recognizer [14].

4.1. Acoustic Features

The acoustic front-ends of the systems consist of MFCCs as base features. The base features are normalized by segment-wise mean and variance and concatenated with a voiceness feature. Furthermore, all feature vectors within a sliding window of length nine are concatenated and projected to a 45 dimensional feature space using a LDA. As proposed in [15, 16] a common LDA for the feature stream is used. Finally, these features are augmented with the PCA transformed probabilistic NN features.

4.2. Acoustic Training

The acoustic models for all systems are based on triphones with cross-word context, modelled by a 3-state left-to-right HMM. A decision tree based state tying is applied resulting in a total of 4500 generalized triphone states. The acoustic models consist of Gaussian mixture distributions with a globally pooled diagonal covariance matrix. Maximum likelihood training is applied and the results presented are from initial experiments without speaker adaptation (SAT/CMLLR) or discriminative training. In [1, 15] we have shown that we still have the same relative improvement after speaker adaptation and discriminative training. Nevertheless, speaker adaptation and discriminative training will be performed next.

The filter banks underlying the MFCC feature extraction undergo a vocal tract length normalization (VTLN). The warping factor classifier is trained beforehand on the complete training corpus, estimated by a grid search in the range of 0.8 - 1.2.

5. Corpora

Approximately 310 hours of American Broadcast News (BN) of speech data are used both for training the probabilistic neural network features and also for training of the acoustic model. The whole corpus consists of 140 hour of HUB4 speech data and 170 hours of TDT4 speech data collected by LDC.

The performance of the different neural network features are estimated on two different kinds of evaluation corpora. In the first experiment the system performance is evaluated on the Broadcast News Transcription development corpus (DEV04) and on the two evaluation corpora EVAL02 and EVAL03. The system parameters have been tuned on the DEV04 corpus. In addition, results on the development corpus (DEV09) and evaluation corpus (EVAL09) of the QUAERO 2009 Evaluation are reported. Again, the development corpus has been used for parameter tuning.

The DEV04 and EVAL03 contain three hour of BN data each and the EVAL02 one hour of BN data. The development and evaluation data of the QUAERO 2009 Evaluation sum up to 15 hours and consist of a mix of different speech sources. The corpora comprise 30 minutes of broadcast news and 30 minutes of speech from the European Parliamentary Plenary Sessions (EPPS) each. The other 11 hours of speech data are collected from the web.

A 4-gram language model (LM) is used in recognition. The LM has been trained on the English Gigaword and HUB4 data and TDT4 data provided by LDC. The lexicon consist of the most frequent 58K entries.

6. Experiments

In order to investigate the relevance of the structure and the input features for a neural network, several systems differing only in the use of the NN probabilistic features, have been set up. Table 1 and Table 2 summarise the results for all corpora. Moreover, systems with NN probabilistic features outperforms the baseline MFCC systems. The relative improvement of over 10% is consistent to the improvement reported by other groups using neural network based features [2, 4].

Table 1: Results of a Gaussian HMM trained systems using MFCCs as base features, augmented with probabilistic features. The probabilistic features are derived from a NN with different topologies and input features.

	Broadcast News		
system	DEV04	EVAL03	EVAL02
MFCC (only)	27.2%	15.8%	13.6%
+ HIER.MRASTA	24.9%	14.1%	12.0%
+ BNECK.TRAP-DCT	23.6%	13.3%	11.8%
+ BNECK.MRASTA	24.3%	13.8%	11.8%
+ HIER-BNECK.MRASTA	22.5%	12.6%	10.5%

Neural Network Input Features: Let us start with the relevance of the input features. As described in Section 2 two different types of input features are used, the TRAP-DCT features and the MRASTA features. As shown in row three

and four of Table 1 the topology of the systems are set fixed to be bottle neck. Even if there is no difference for the EVAL02 corpus, the BNECK.TRAP-DCT features outperform the BNECK.MRASTA for all other corpora. In case of DEV04 and EVAL03 the system performance has been improved by about 3%-4% relative. This improvement is also consistent for the QUAERO corpora in Table 2. Since there is only a small improvement for EVAL09, MRASTA features seem to generalise much better than the TRAP-DCT features. Further investigations have to be done to verify the performance of the different features.

Table 2: Results of a Gaussian HMM trained systems using MFCCs as base features, augmented with probabilistic features. The probabilistic features are derived from a NN with different topologies and input features. Experiments have been done on the QUAERO 2009 development and evaluation corpus.

	QUAERO 2009	
system	DEV09	EVAL09
MFCC (only)	52.1%	42.9%
+ HIER.MRASTA	48.5%	39.1%
+ BNECK.TRAP-DCT	46.9%	39.4%
+ BNECK.MRASTA	48.5%	39.4%
+ HIER-BNECK.MRASTA	45.7%	37.8%

Neural Network Topology: The other change in the NN based feature extraction has been the topology of the neural network itself. As shown in row two (HIER.MRASTA) and four (BNECK.MRASTA) of Table 1 the results are slightly different only. The difference from the DEV04 corpus could not be verified on the evaluation corpora. Moreover, in Table 2 the behaviour has been turned in the opposite direction. Overall, none of the concepts could outperform the other.

Nevertheless, in order to benefit from both approaches we have connected the two topologies. The results are given in the last row of Table 1 and Table 2 (HIER-BNECK.MRASTA). The best result using MRASTA features could be improved by up to 6%-7% relative for broadcast news and even the previously best result using the TRAP-DCT features could be improved by 3%-5% relative for all corpora.

Overall, the single MFCC system could be improved by about 12% relative for the QUAERO corpora and 17% - 23% relative to the other corpora.

7. Summary and Conclusions

In this paper, we investigated different neural network topologies such as hierarchical processing and bottle neck features for neural network based feature extraction. Since none of the hierarchical processing or bottle neck processing could outperform each other we suggested a hierarchical bottle neck processing. This new topology for NN feature extraction could benefit from both approaches and had combined the advantages. Depending on the corpora the hierarchical bottle neck features improved the system by 3% - 7% relative. Moreover, the single MFCC based system was improved by up to 20% relative.

In addition, we had analysed two different types of input feature where the TRAP-DCT based feature extraction has performed best. Next, we will combine the TRAP-DCT features and the hierarchical bottle neck processing to further improve the feature extraction. In order to get the maximum gain from the hierarchical bottle neck processing we will spilt up the TRAP-DCT features similar to the MRASTA filtering. Furthermore, a complete training, including speaker adaptation and discriminative training will be performed.

8. Acknowledgements

This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

The Authors would thank IDIAP for the support, especially Fabio Valente.

9. References

[1] C. Plahl, B. Hoffmeister, G. Heigold, J. Lööf, R. Schlüter, and H. Ney, "Development of the GALE 2008 Mandarin LVCSR system," in *Interspeech*, Brighton, U.K., Sep. 2009, pp. 2107–2110.

[2] X. Lei, W. Wu, W. Wang, A. Mandal, and A. Stolcke, "Development of the 2008 SRI Mandarin speech-to-text system for broadcast news and conversation," in *Interspeech*, Brighton, U.K., Sep. 2009, pp. 2099–2103.

[3] F. Valente, M. Magimai-Doss, C. Plahl, S. Ravuri, and W. Wang, "A comparative large scale study of MLP features for Mandarin ASR," in *Interspeech*, Makuhari, Japan, Sep. 2010.

[4] Q. Zhu, B. Chen, N. Morgan, and A. Stolke, "On using MLP features in LVCSR," in *Interspeech*, Jeju Island, Korea, Oct. 2004.

[5] H. Hermansky and S. Sharma, "TRAPs - classifiers of temporal patterns," in *Proc. Int. Conf. on Spoken Language Processing*, Sydney, Australia, Dec. 1998.

[6] H. Hermansky and P. Fousek, "Multi-resolution RASTA filtering for TANDEM-based ASR," in *Interspeech*, Lisbon, Portugal, Sep. 2005, pp. 361–364.

[7] H. Hermansky, D. Ellis, and S. Sharma, "Tandem connectionist feature stream extraction for conventional HMM systems," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2000, pp. 1635–1638.

[8] F. Valente, J. Vepa, C. Plahl, C. Gollan, H. Hermansky, and R. Schlüter, "Hierarchical neural networks feature extraction for LVCSR system," in *Interspeech*, Antwerp, Belgium, Aug. 2007, pp. 42–45.

[9] F. Grézl, M. Karafiat, S. Kontar, and J. Cernock, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 4, Honolulu, HI, USA, Apr. 2007, pp. 757–760.

[10] F. Grézl and P. Fousek, "Optimizing bottle-neck features for LVCSR," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, USA, Apr. 2008, pp. 4729–4732.

[11] F. Grézl and H. Hermansky, "Local averaging and differentiating of spectral plane for TRAP-based ASR," in *Interspeech*, Geneva, Switzerland, Sep. 2003, pp. 1017–1020.

[12] F. Valente, M. Magimai.-Doss, C. Plahl, and S. Ravuri, "Hierarchical processing of the modulation spectrum for GALE Mandarin LVCSR system," in *Interspeech*, Brighton, U.K., Sep. 2009, pp. 2963–2966.

[13] M. Nußbaum-Thom, S. Wiesler, M. Sundermeyer, C. Plahl, S. Hahn, R. Schlüter, and H. Ney, "The RWTH 2009 QUAERO ASR evaluation system for English and German," in *Interspeech*, Makuhari, Japan, Sep. 2010.

[14] D. Rybach, C. Gollan, G. Heigold, B. Hoffmeister, J. Lööf, R. Schlüter, and H. Ney, "The RWTH Aachen University open source speech recognition system," in *Interspeech*, Brighton, U.K., Sep. 2009, pp. 2111–2114.

[15] C. Plahl, B. Hoffmeister, M.-Y. Hwang, D. Lu, G. Heigold, J. Lööf, R. Schlüter, and H. Ney, "Recent improvements of the RWTH GALE Mandarin LVCSR system," in *Interspeech*, Australia, Aug. 2008, pp. 2426–2429.

[16] R. Schlüter, A. Zolnay, and H. Ney, "Feature combination using linear discriminant analysis and its pitfalls," in *Interspeech*, Pittsburgh, PA, USA, Sep. 2006, pp. 345–348.