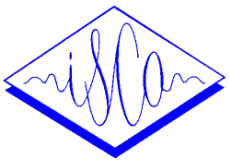

3rd European Conference on
Speech Communication and Technology
EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

SYLLABLE SEGMENTATION OF CONTINUOUS SPEECH WITH ARTIFICIAL NEURAL NETWORKS

W. Reichl and G. Ruske

Lehrstuhl für Datenverarbeitung, Technische Universität München
Franz-Joseph-Str. 38, D-80801 München, Germany

ABSTRACT

This paper describes the utilization of multilayer perceptrons and radial basis function networks for the syllable segmentation of continuous speech by detecting the syllable nuclei. The artificial neural nets were trained to indicate the occurrence of vowels or diphthongs by means of the backpropagation algorithm and a non-iterative matrix-inversion method. In different experiments the syllable nuclei were correctly marked with more than 93 % and less than 2 % insertions.

Keywords: Syllable segmentation, neural networks, multilayer perceptrons, radial basis functions

10.21437/Eurospeech.1993-182

1. INTRODUCTION

For many tasks in automatic speech recognition a reliable segmentation of continuous speech would be very helpful. For instance, segmentation information can be used in order to reduce search space (pruning). The segmentation of words or subword units in continuous speech is a rather difficult task and one possible way consists in utilising the syllable structure of speech. This leads to a consonant sequence between two successive syllable nuclei, the latter of which consist of a vowel or a diphthong. To derive a syllable segmentation of continuous speech we therefore need a detector marking the syllable nuclei.

Artificial neural nets showed high performance in many classification tasks [1]. Their capability of modeling nonlinear transformations and their simple training algorithm makes them attractable for complex feature detector design. Therefore different kinds of artificial neural nets, multilayer perceptrons, and nets with radial basis functions, were trained to indicate the occurrence of vowels or diphthongs. This was performed by the backpropagation algorithm and a non-iterative matrix-inversion method. Details of the training algorithm for the radial basis nets and the results of various experiments with different nets will be explained in the next sections of this paper.

2. DATABASE

In the experiments for the syllable segmentation of continuous speech we used a speaker independent database of 10 German speakers (Phondat: 'Berliner'-sentences). Each speaker uttered 2 versions of 100 German sentences. We used the utterances of 6 speakers for training (dataset I) and the remaining 4 speakers for testing the generalisation capability of the neural networks (dataset II). For some experiments we checked the performance of the trained detectors with a third independent database (dataset III). This consists of another 2 versions of different 100 German sentences, spoken by 6 of the 10 speakers (training and test speakers) from the first database (Phondat: 'Marburger'-sentences).

The speech signals were sampled at 16 kHz and quantized with 16 bit. After a 256-point FFT with Hamming window the power spectrum was comprised in critical bands. In this way we got every 10 ms a Bark-scaled loudness spectrum, which was normalized to sum up to one. In some experiments additional features as the total loudness, the delta-loudness spectrum and the zero-crossing rate of the signal were utilized.

3. MULTILAYER PERCEPTRONS

In our first experiments we used feed-forward fully-connected multilayer perceptrons (MLP) with different numbers of hidden layers (one and two hidden layers) and various numbers of nodes in the hidden layers. The input layer of the MLP net is made up of a sliding window of several frames. Each frame consists of different features derived from the acoustic processing every 10 ms. The syllable nuclei were indicated by the single node in the output layer. We trained the nets using the backpropagation algorithm and adapted the weights after the presentation of every pattern (stochastic approximation). The training was stopped when the performance on the test data decreased (cross-validation). Because we only wanted the neural nets to indicate the position of the syllable nuclei somewhere within the vowel or diphthong range, we eval-

uated the contour of the output activation for the position of the syllable nuclei, by using the maximum values of the smoothed activation. Only distinct maxima which exceed a threshold in their activation and which were separable from their neighbours were used as syllable nuclei. In Figure 1 an example of the activation of the output node for the German sentence "*Heute ist schönes Frühlingswetter* " is given. In addition the target values for the training of the neural net are depicted in Figure 1. The vertical lines mark the detected positions of the syllable nuclei.

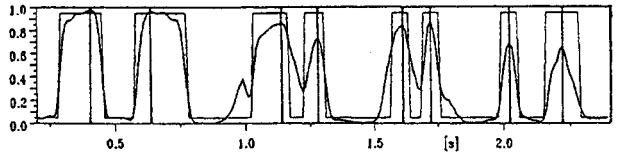

Figure 1: Activation of the output node for the German sentence "*Heute ist schönes Frühlingswetter* "

In Table 1 we summarized the results of different experiments with the multilayer perceptron nets. In the first column the numbers of neurons in the hidden layers are given. We used 5, 10 and 20 neurons in nets with one hidden layer and 10 neurons in the first and 4 neurons in the second hidden layer in nets with two hidden layers. The next two columns of Table 1 show the recognition rate of correctly detected syllable nuclei and the insertion rate of additionally indicated ones. All values are in %. A syllable nucleus is registered as a correct one if he is marked somewhere within the vowel or diphthong range. The results of multilayer perceptrons trained with a feature vector consisting of the 20 dimensional loudness spectrum, the total loudness and the zero crossing rate are printed in column 2. The MLP input layer is made up of a sliding window of 5 consecutive frames of the 22 dimensional feature vector and has in total 110 nodes. The results of the nets which additionally utilize the delta-loudness spectrum are depicted in column 3. This feature incorporates information about the temporal process of the spectrum and hence, no temporal window in the input layer (1x42 neurons) was used. The recognition and insertion rates in % for the training data - dataset I - are given in the first lines for all the nets in Table 1. The results for the cross reference test - dataset II - and the results for the second test set - dataset III - are illustrated, too.

All the nets with a temporal window of 5 frames worked very well. They correctly indicated up to 94.5 % of the syllable nuclei of the training and 93.7 % of the cross-reference data. The detectors additionally marked approx. 1.8 % syllable nuclei in the training set. The insertions in the test set are below 1 %. Subsequently we checked the performance of the neural nets with the third dataset. The results of all the nets are better, up to 98.5 % detection rate, which indicates a simpler task for this material. All the rates for the nets without a temporal context (column 3 in Table 1) show about 1 % to 2 % inferior values. The detection rates are below 93.3 % for the training, 93.6 % for the cross-validation and 97.3 % for the test sets. The

Nr. Neurons Hidden Layer	Input Layer 5x22		Input Layer 1x42		Data
5	93.9	1.8	91.8	3.0	I
	93.7	1.0	92.5	0.8	II
	98.1	2.0	97.1	3.4	III
10	94.5	1.9	93.3	2.8	I
	93.7	0.8	93.6	1.4	II
	98.3	2.3	97.3	3.6	III
20	93.4	1.8	91.0	3.1	I
	92.7	0.6	91.1	1.2	II
	98.1	2.3	97.2	3.5	III
10,4	92.5	2.0	90.8	1.8	I
	91.7	0.9	89.6	1.4	II
	98.5	1.8	97.0	2.0	III

Table 1: Recognition rates and insertion rates for different MLPs; all values in %

information about the temporal process of the spectrum, incorporated in the delta-loudness, is inferior to the possibility of adjusting the weights to the frames in the temporal window. A detailed analysis of the detection errors shows that most problems with insertions occur with the vowel-like consonants /l/ and /R/. Most deletions of syllable nuclei arrise for the schwa-sound /@/ and the vowel /Y/.

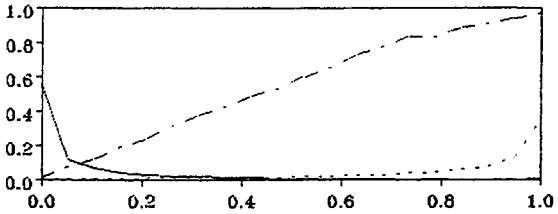

Figure 2: Histogram of the output activation for vowel and non-vowel frames and the computed a-posteriori propability of a syllable nucleus conditioned by the output

In [1] it was shown that neural nets, trained with the mean squared error (MSE) criterion, approximate on their output activations the a-posteriori propability $p(c|x)$ of class c depending on input x. A neural net reaching its global minimum of the error criterion results in an optimal Bayes classifier. Histograms of the output activation for vowel and non-vowel input data were computed to examine if the output activations of the trained nets approximate the propability of a syllable nucleus conditioned by the input data. The solid line in Figure 2 shows the distribution $p(out|A)$ of the output activation of non-vowel (class A) and the dotted line the distribution $p(out|B)$ of vowel frames (class B). The separation of both classes is quite good. According to Bayes rule the propability $p(B|out)$ of a syllable nucleus depending on the output activation is

$$p(B|out) = \frac{p(out|B)p(B)}{p(out|A)p(A) + p(out|B)p(B)} \qquad (1)$$

The dashed line in Figure 2 displays the a-posteriori propability $p(B|out)$. For all the trained neural nets we found the diagonal in the histograms: $p(B|out) \approx out$. For this reason we can maintain that the outputs of the multilayer perceptrons, trained on the mean squared error criterion to detect syllable nuclei, really approximate the conditioned propability of a vowel or diphthong.

4. NEURAL NETS WITH RADIAL BASIS FUNCTIONS

A disadvantage in working with multilayer perceptrons is the initialization of the weights with random values. Just after a long training period the net parameters are usefully adjusted. A better way seems to use locally tuned neurons and initialize the neuron centers and ranges carefully [3]. In contrary to MLP nets the neurons in the first hidden layer of a radial basis function net compute the weighted distances from the input vector $\vec{x}$ to their means $\vec{m}$. We use the Mahalanobis distance which is exponentially weighted by the nonlinear transfer function of the neurons. The activation of such a neuron j is

$$y_j = \exp\left(-(\vec{x} - \vec{m}_j)^T C_j^{-1} (\vec{x} - \vec{m}_j)\right) \quad (2)$$

with the covariance matrix C_j. All the activations of the neurons $y_j(t)$ for one pattern t are collected in $\vec{y}(t)$. One advantage of such neurons consists in the initialization of mean and covariance by a clustering method [3]. We used the LBG algorithm [2] to determine the initial positions of the hidden neurons. These neurons describe the feature space by means of hyperellipses, like Gaussian functions, instead of hyperplanes for MLPs. The next layer of neurons in the RBF net is just like in MLPs. The nodes compute a weighted sum of the activations of the preceeding layer and use a transfer function $f(a)$ to determine their output values

$$out = f(\vec{y}^T \vec{w}) \quad (3)$$

Sometimes the transfer function is just linear $f(a) = a$ as in [3,4] and the problem of finding the optimal weight vector $\vec{w}$ for fixed hidden neurons becomes the solution of a linear system. This can be done non-iteratively by a matrix inversion method leading to the optimal solution with respect to the mean squared error. If Y is the matrix of the activations of the hidden neurons $Y = [\vec{y}(1), \cdots, \vec{y}(T)]$ for all the patterns $t = 1, \cdots, T$ the weights derives as

$$\vec{w} = (YY^T)^{-1} Y \vec{z} \quad (4)$$

$(YY^T)^{-1}Y$ is called the pseudo-inverse of Y and $\vec{z}$ is the vector of the target values for all patterns [4].

In case of sigmoid transfer functions $f(a) = \frac{1}{1+\exp(-a)}$ an iterative training algorithm, e.g. a gradient descent method, is necessary to adjust the weights of the sigmoidal units as well as the means and covariances of the radial basis functions. The required partial derivations for a mean squared error criterion $E = \frac{1}{2}(z - out)^2$ are

$$\frac{\partial E}{\partial \vec{w}} = -(z - out) f' \vec{y} \quad (5)$$

$$\frac{\partial E}{\partial \vec{m}_j} = -(z - out) f' w_j y_j C_j^{-1} (\vec{x} - \vec{m}_j) \quad (6)$$

$$\frac{\partial E}{\partial C_j^{-1}} = -(z - out) f' w_j y_j \left[-(\vec{x} - \vec{m}_j)(\vec{x} - \vec{m}_j)^T\right] \quad (7)$$

f' is the derivation of the sigmoid function to its argument $f'(a) = f(a)(1 - f(a))$. The resulting training algorithm is the same as the backpropagation algorithm for MLPs and iteratively adjusts the parameters of the net in order to decrease the error criterion.

We first used radial basis nets with one layer of basis neurons and a linear node in the output layer. The centers and ranges of the basis functions were derived from the LBG algorithm and kept fixed. For sake of simplification we used only diagonal covariance matrices. The weights of the output node were computed via the pseudo-inverse method. Preliminary experiments showed some problems with the linear structure used in the output layer. This kind of net is often used in neural network approaches for function approximation [3], but has not the capability of handling the dynamics of the hidden neuron activation. For classification purposes we prefer an output activation of 1 for all vowel input data, but with this linear weighting scheme there is no way to compensate the rapid change of the hidden activations during the course of a vowel. For this purpose the variances were increased to make the hidden unit activation more smooth when a vowel proceeds. Some combinations of the features loudness spectrum (LOUD), total loudness (TOTAL), zero crossing rate (ZERO) and delta-loudness (DELTA) were used in the experiments. In Table 2 the numbers of basis neurons for these features are given. The LBG algorithm was used separately for the distinct features. A temporal context of 5 frames in the hidden layer resulted in connections from the output neuron to the activities of the hidden nodes within 5 frames (see Figure 3).

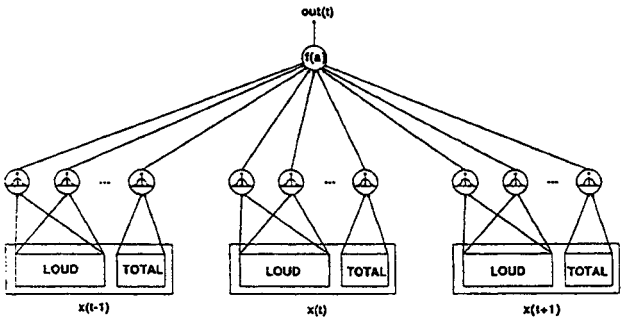

Figure 3: RBF net with 2 features and a symmetric temporal context of 3 frames

The detection and insertion rates for the same task as for the MLPs are shown in Table 2 for the RBF nets. The results for dataset I and II are inferior to these of Table 1. The simplest net works best, detecting 92.1 % of the syllable nuclei in the training data and 89.2 % in the test data with about 3.5 % insertions.

Nr. Neurons						Data
LOUD	TOTAL	ZERO	DELTA			
16	16	0	16	92.1	3.6	I
				89.2	3.5	II
16	0	16	16	90.3	7.9	I
				87.5	8.5	II
64	0	64	64	89.8	6.5	I
				87.1	6.0	II

Table 2: Recognition rates and insertion rates for different RBF nets; all values in %

The net with 64 neurons for each feature was choosen for an additional experiment for the adjustment of the centers of the basis functions. An additional nonlinear sigmoid function was added in the output node, because a nonlinear transformation of the hidden neuron activations could handle the occurring dynamics of the hidden neurons. A gradient descent method was used for the fine-tuning of the weights and means of the net. With this modifications an improvement in the performance, up to 94.3 % detection rate for training and 88.4 % for testing data, was registered. Because the basis centers were not adjusted significantly by the gradient descent training this was mainly dedicated to the nonlinear structure in the output layer. We therefore decided to fix the means of the basis functions again and to add an additional nonlinear layer of neurons. Since the experiments with the MLPs showed better results with a temporal context than using the delta-loudness spectrum no delta-loudness was used anymore. The RBF nets for the next experiments contained different numbers of basis neurons, a temporal context in the hidden layer with 5 or 10 sigmoid neurons and a sigmoid output neuron (see Table 3).

Nr. Neurons								Data
			Hidden Layer					
LOUD	TOTAL	ZERO	5		10			
16	16	0	94.9	2.1	95.2	2.1	I	
			92.0	2.7	92.0	2.6	II	
			95.4	1.8	95.7	2.0	III	
128	16	0	96.1	1.9	96.7	1.9	I	
			89.8	3.6	90.5	4.0	II	
			96.6	2.0	96.9	2.1	III	
16	16	16	94.7	2.0	95.1	2.2	I	
			90.7	2.9	92.3	3.0	II	
			94.2	1.8	95.3	2.2	III	
128	16	16	96.8	1.9	97.0	1.9	I	
			91.3	4.3	91.0	4.8	II	
			97.0	2.5	97.1	2.3	III	
256	16	16	96.2	1.9	96.1	1.9	I	
			90.1	5.7	89.7	5.0	II	
			96.3	2.3	96.5	2.2	III	

Table 3: Recognition rates and insertion rates for RBF nets with an additional nonlinear layer; all values in %

The nets with 10 hidden neurons show only minor improvements compare to the nets with 5 neurons. The utilization of 128 basis neurons for the loudness results in an improvement in the detection rate for the training data (I) but in a decreased performance in test data (II). The rates for nets with 256 basis neurons show some degeneration, because the description of the feature space is too precise for this task. The best results for the training data are 97.0 % detection and 1.9 % insertion rate for the net with 128,16,16 basis neurons and 10 hidden neurons. The simpler nets with fewer basis neurons yielded better rates for the test set: 92.3 % detection and 3.0 % insertion rate. The results for test set III are, just as in the MLP experiments, roughly the same as for the training set. In comparision to the MLP the training rates are higher, which is attributed to the more complex structure of the nets with much more adjustable parameters. These were adapted to better fit on the training data and resulted in a decreased generalisation performance.

5. CONCLUSION

Our experiments show good results with all our MLPs and RBFs, independent of the numbers of hidden layers and hidden nodes. The best generalisation performance was derived from the simple MLP nets. More than 93 % of the syllable nuclei were detected correctly with less than 2 % of insertions in a speaker independent task with difficult speech material. This shows the applicability of the neural net approach to such a feature detection task. The more complex RBF nets result in higher performance in the reclassification test for the training data, but in decreased generalisation capability. The structures of the nets are too complex for this task and the training algorithm results in overadaptation of the parameters. The particular advantage of the RBF structure, the initialization procedure for the means and variances, brings some speed-up in computation time.

REFERENCES

[1] *H.Bourlard, C.J.Wellekens:* Links Between Markov Models and Multilayer Perceptrons, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 12, No. 12, S. 1167-1178, December 1990.

[2] *Y.Linde, A.Buzo, R.Gray:* An Algorithm for Vector Quantiser Design, IEEE Trans. on Communications, Vol. 28, No. 1, S. 84-95, January 1980.

[3] *J.Moody, C.J.Darken:* Fast Learning in Networks of Locally-Tuned Processing Units, Neural Computation, Vol. 1, No. 2, S. 281-294, 1989.

[4] *S.Renals, R.Rohwer:* Phoneme Classification Experiments Using Radial Basis Functions, Proc. of the Int. Joint Conf. on Neural Networks, Washington, D.C., Vol. 1, S. 461-467, June 1989.