A STUDY OF PERCEPTRON MAPPING CAPABILITY TO DESIGN SPEECH EVENT DETECTORS

Sabato M. Siniscalchi[1,2], Mark A. Clements[1], Antonio Gentile[2], Giorgio Vassallo[2], and Filippo Sorbello[2]

[1]School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332, USA
[2]Dipartimento di Ingegneria Informatica
Università degli Studi di Palermo, Palermo 90128, Italy
{marco, clements}@ece.gatech.edu, {gentile, gvassallo, sorbello}@unipa.it

ABSTRACT

Event detection is a fundamental yet critical component in automatic speech recognition (ASR) systems that attempt to extract knowledge-based features at the front-end level. In this context, it is common practice to design the detectors inside well-known frameworks based on artificial neural network (ANN) or support vector machine (SVM). In the case of ANN, speech scientists often design their detector architecture relying on conventional feed-forward multi-layer perceptron (MLP) with sigmoidal activation function. The aim of this paper is to introduce other ANN architectures inside the context of detection-based ASR. In particular, a bank of feed-forward MLPs using sinusoidal activation functions is set up to address the event detection problem. Experimental results demonstrate the effectiveness of this ANN design for speech attribute detectors.

1. INTRODUCTION

Recently, many attempts have been made to inject articulatory information into Hidden Markov Models (HMMs). All of these attempts are driven by two conjectures: articulatory features are more robust for noise and cross-speaker variation, and they are easily represented in the acoustic domain. Furthermore, standard acoustic features, such as Mel-Frequency Cepstrum Coefficients (MFCCs) and articulatory features are defined in domains sufficiently different to carry complementary sources of information when combined and prove very useful for automatic speech recognition tasks [2]. Speech scientists often infer articulatory information by statistical methods. For example in [1], [2], [3], [4] standard MFCC-based vectors are the input of artificial neural networks (ANN) whose output simulate the posterior probabilities of certain events, such as manner and place of articulation. In [5] support vector machine based classifiers have been employed to classify manner of articulation events.

These paradigms, in which external knowledge is combined with the classical ASR approach, are usually referred to as knowledge-based paradigms. The key idea is to build a bank of speech event detectors which gives consistent detection results. An event can be thought of as a low level speech attribute that carries a piece of information required to form higher level "evidences," useful for phone, word, and sentence detection. Nevertheless, as pointed out in [6], event detection is a challenging task, and it is usually more complicated than signal detection. This difficulty is mainly due to the fact that events do not rely on as well-defined a theory as does signal detection [7], and also to the wide variation in event duration, which can range from few milliseconds to seconds. As a result, event detection is a critical component of most knowledge-based paradigm approaches, and the design of high-performance speech event detectors is a challenging task.

This paper is focused on the design of "non-conventional" learning machines as a basis for event detectors. Different ANN designs are evaluated to achieve robust speech event detectors. The shape of the classification boundary of ANN with sinusoidal activation function [8], and high order neurons [9] will be studied and compared with standard sigmoidal-based ANNs. Attention will be focused specifically on manner of articulation features, namely vowel, fricative, stop, nasal, approximant, and silence. Experimental results on the TIMIT database demonstrate the effectiveness of the proposed artificial neural network design for speech attribute detection.

The rest of the paper is organized as follows. Section 2 describes detector design using ANN. Section 3 illustrates experimental setup and results achieved. Section 4 concludes the paper.

2. ANN-BASED DETECTOR DESIGN

It is well known that artificial neural networks can be widely used in the field of pattern recognition, since they can learn a mapping from an input space to an output space. They can realize a compromise between recognition speed, recognition rate, and hardware resources [10]. Moreover, the generalization capability of neural networks is acquired during the training phase, and the generalization degree achieved is strictly related to the training set characteristics. A great number of parameters must be taken into account when working with these connectionist models, such as the number of hidden layers, the number of processing units in each layers, and the shape of the activation function of the processing unit. The latter parameter is by far the most important one since its nature determines the mapping from the input to the output domain. Hence, a wrong choice may invalidate all of the

effort made to accomplish the desired task. Gori and Tesi in their work [10] state that by using linear functions the backpropagation algorithm reaches the global minimum of the error function if the classes are linearly separable. However, the choice of linear functions opposes the proof that a neuron transformation is based on a nonlinear function. Allen and Stork [11] have postulated the constraint that the activation function should grow monotonically. In this direction, they carried out a study on the choice of the activation function that leads good results in terms of classification. Common examples of such activation functions are the sign function and the family of sigmoids. Because of their monotonically increasing characteristics, however, all of those functions suffer from the drawbacks of slow convergence and poor nonlinear mapping ability. Relaxing the monotonic constraint can provide additional computational power as evidenced by the Gaussian activated value units of Dawson and Schopflocher [12].

In light of this discussion, the authors believe that the choice of the processing units of the hidden layers of ANNs is very critical if good detection rate is to be reached in the task at hand. Consequently, the two following sections investigate the nature of processing unit that can lead to a better global minimum of the error function.

2.1. Sinusoidal Activation Function

Although many ways exist to simulate the non-monotonic behavior, a simple choice is the sinusoid function with unit amplitude [8]. Its output will be a periodic function of its input in the range [-π, π]. It was found [13] that the Vapnik-Chervonesnkis (VC) dimension of the sinusoidal function is greater than that of the sigmoid function. Consequently, the nonlinear mapping capability is improved and the convergence is faster.

A gradient descent technique can be used in the training phase since this function is both continuous and differentiable throughout its domain. Moreover, the power of the processing unit can be inferred by considering the number of decision regions that that the unit induces onto its input domain. Figure 2.1 gives a conceptual picture of the manner in which the periodic, sigmoid, and Gaussian unit carves decision regions in a 2-dimensional input space. Intuitively, it can be stated that the larger the number of induced decision regions is, the smaller the number of processing units are that are needed to accomplish a given task. Alternatively, the task can be accomplished in less time. Empirical experiments have validated these intuitions [8].

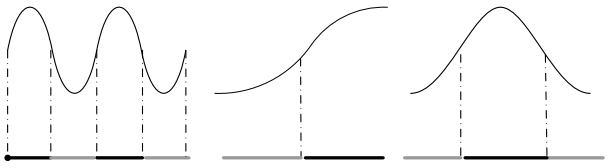

Figure 2.1. Partitioning induced by sinusoid, sigmoid, and Gaussian activation function onto the input domain.

To gain more clues about the power of the sinusoidal activation function, it may be insightful to examine the XOR problem. It is well know that that problem can not be resolved by means of a single perceptron with only one hidden unit using a sigmoidal activation function or with any growing monotonic function. Such a function $f(I_1 \cdot w_1 + I_2 \cdot w_2 + \theta)$ would not classify the input points ($I_1 = 1$, $I_2 = 0$) and ($I_1 = 0$, $I_2 = 1$) with a high output value and the input ($I_1 = 1$, $I_2 = 1$) with a low output value.

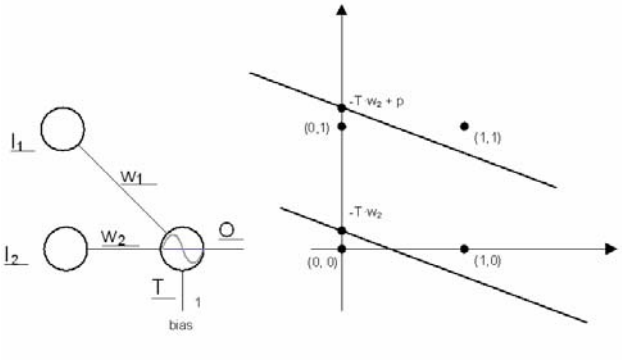

Figure 2.2. Perceptron employed for the XOR problem (left); separation boundary of the two classes with sinusoidal function (right).

Instead, if a sinusoidal function is chosen as the activation function, the above-mentioned conditions can be satisfied thanks to the periodicity of this function. Moreover, it is easy to prove that a sinusoidal function separates the classes by parallel straight lines and, in the case of multidimensional inputs, by parallel hyperplanes. The proof is given below. For the XOR problem, in fact, the general expression of the activation function is given in equation (2.1).

$$O(\bar{I}, \bar{w}) = \sin(I_1 \cdot w_1 + I_2 \cdot w_2 + \theta) \ , \qquad (2.1)$$

The separation boundary between the two classes is obtained from equation (2.1) by imposing the condition given in equation (2.2).

$$I_1 \cdot w_1 + I_2 \cdot w_2 + \theta = k\pi \quad k = \pm 1, \pm 2, ... , \qquad (2.2)$$

On the input space, this represents, the equation of parallel straight lines with angular coefficient $-w_1 / w_2$ (see Figure 2.2).

2.2. Hyperspherical Neuron

The perceptron's job with a sigmoid activation function is to separate the pattern space by a hyperplane. When the input classes are not linearly separable, many neurons are put together in a particular structure so that the combination of their linear decision planes can approximate non-linear surfaces. Nevertheless, if the input data has a particular structure it may be convenient to use neuron with non-linear decision surfaces, such as hyperbolic, or hyperspherical ones. In this context, the latter is presented next. The hyperspherical neuron is a special kind of what are called higher order neuron [9]. The main concept behind high order neuron is to employ a non-linear decision boundary to classify the input patterns, and the leading conjecture is that the more complex the decision boundary is, the fewer number of neurons is to accomplish a given task. In the case of the hyperspherical neuron the decision surface is a hypersphere, which becomes a sphere in the two dimensional space. Although there are many ways to interpret the hyperspherical processing unit, the Clifford algebra

[15] representation seems to be the more straightforward and useful. The key idea is to embed the Euclidean space in a conformal space in which it is possible to compute the scalar product of a point and sphere and verify whether the point falls inside or outside the hypersphere. The mathematical formulation of the procedure to embed the Euclidean space into a conformal space follows [15], although in here only the part useful for the scope of this paper is presented.

The starting point to build the conformal space is the Minkowski plane $R^{1,1}$, which has an orthonormal basis $\{e_+, e_-\}$ whit $(e_\pm)^2 = \pm 1$, and $e_+ \cdot e_- = 0$. In this plane a null basis $\{e_0, e_\infty\}$ can be defined by the pair of equations given in (2.3),

$$e_0 = \frac{1}{2}(e_- - e_+), \text{ and } e_\infty = e_- + e_+ \qquad (2.3)$$

where $(e_0)^2 = (e_\infty)^2 = 0$ and $e_0 \cdot e_\infty = -1$. The conformal space ME^n can be generated by direct sum of the R^n Euclidean space and the $R^{1,1}$ Minkowski plane as given in equation (2.4).

$$ME^n = R^{n+1,1} = R^n \oplus R^{1,1}, \qquad (2.4)$$

To express the R^n vectors in the ME^n it is required that the former are the null vectors in the conformal space. As a result, the set of R^n **x** belongs to the horosphere in the conformal space described by (2.5).

$$\left\{X \in ME^n \mid X^2 = 0, X \bullet e_\infty = -1\right\}, \qquad (2.5)$$

From to conditions $X^2 = 0$ and $X \cdot e_\infty = -1$, the **x** vector may be projected onto the conformal space by the transformation formula (2.6).

$$X = x + \frac{1}{2}x^2 e_\infty + e_0, \qquad (2.6)$$

$$X \cdot Y = -\frac{1}{2}(x - y)^2, \qquad (2.7)$$

It easy to see that the scalar product of the vector X time the vector Y in ME^n gives the Euclidean distance of the corresponding R^n points, as shown in equation (2.7). Moreover, since $S = Y - (1/2)r^2 e_\infty$ is a normalized hypersphere with center in the null vector Y and radius r, the scalar product of a null vector X and S is given in equation (2.8). This product will be positive if X is inside the hypersphere; negative if it is outside the hypersphere; and null if it is on the hypersphere.

$$X \cdot S = X \cdot Y - \frac{1}{2}r^2 X \cdot e_\infty = -\frac{1}{2}(x - y)^2 + \frac{1}{2}r^2, \qquad (2.8)$$

This change is useful in the process of building a hyperspherical processing unit in the context of pattern classification.

In the realization of the hyperspherical neuron the guidelines given in [9] were followed. Hence, a R^n data vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ is transformed into a R^{n+2} data vector $\mathbf{X} = (x_1, x_2, \ldots, x_n, -1, -(1/2)\mathbf{x}^2)$, and a generic ME^n hypersphere $S = \mathbf{y} + (1/2)(\mathbf{y}^2 - r^2)\, e_\infty + e_0$ as $\mathbf{S} = (y_1, y_2, \ldots, y_n, (1/2)(\mathbf{y}^2 - r^2), 1)$. Finally, it is important to point out that a hypersphere with infinite radius becomes a hyperplane, so the conventional neuron could be thought as a special case of the hyperspherical neuron.

3. EXPERIMENTS

An evaluation of the sigmoid, sinusoid, and hypersphere detectors was performed on the TIMIT database [16]. For fair comparison with other results, the training and the test sets are the same as used in [2]. Consequently, the utterances used for speaker adaptation (SA) were excluded. 3696 utterances were used for the training phase and 192 utterances for the testing phase. Furthermore, the ANN was trained on only 3504 randomly selected utterances out of the initial 3696. Three different sets of six detectors were designed to detect six manner of articulation events, namely vowels, fricatives, nasals, stops, approximants, and silence. Within each bank of detectors the ANNs share the same structure. In each of the three sets a different neuron, the same topology is used, namely sigmoid activation function, sinusoid activation function, and hypersherical neuron. To be consistent with [2], no parameter tuning was performed, the ANNs were trained in a frame-based fashion, and 200 iterations were performed. Each of the detectors was designed as feed-forward multi-layer perceptron with 117 inputs, 100 hidden nodes in the hidden layer, and two outputs with linear activation function. The dimension of the input is obtained by concatenating the current frame with four preceding frames and four following ones, so that each input vector represents nine frames. A single frame is compounded by 12 Mel-Frequency Cepstrum Coefficients and the logarithm of the Energy. The frame-rate is set at 10 ms.

%	Sigmoid	Sinusoid	Hypersphere
Vowel	11.29	**8.67**	14.07
Fricative	6.50	**4.32**	7.2
Stop	8.31	**5.25**	9.34
Nasal	4.95	**3.15**	6.66
Approximant	8.91	**6.22**	10.61
Silence	1.81	**1.30**	1.92

Table 1. Average error rate (%) achieved by each detector.

Table 1 gives the average error rate achieved by each of the three detector banks over all of the experiments conducted. The error rate is computed as the ratio between the number of misclassified frames over the total number of frames. The results show an average of about 2% absolute improvement achieved using the sinusoid activation function with respect to the sigmoid one on all six events. On the other hand, Table 1 also shows that hyperspherical detectors do not perform well for manner of articulation events.

%	Vowel	Fricative	Stop	Nas.	Appr	Sil.
Vowel	**91.00**	1.38	1.53	1.26	4.64	0.19
Fricative	3.16	**88.06**	5.53	1.02	0.89	1.24
Stop	6.32	7.41	**81.03**	1.71	1.57	1.96
Nasal	9.65	2.44	3.25	**81.45**	2.20	0.90
Approximant	30.82	2.88	3.26	2.74	**59.11**	1.19
Silence	1.10	1.09	1.88	0.61	0.58	**94.74**

Table 2. Confusion matrix of sinusoidal detectors for the manner of articulation events

The results obtained thus far offer evidence that stresses the importance of selecting the proper activation function for the specific task at hand. This is further emphasized by the common practice in the speech community to use the output of each detector as a score to inject external knowledge into standard ASRs. To compare the output for the six detectors, a global

confusion matrix is built by considering as winner the unit with the highest output value. This matrix is given in Table 2. It is observed a 3.5% absolute improvement over the baseline [2], as shown in Table 3, with an 8.5% absolute improvement for the stop class. Further improvements might be achieved by tuning learning and architecture parameters.

%	Vowel	Fricative	Stop	Nas,	Appr	Sil.
Actual Class	2.00	2.86	8.53	3.95	2.61	1.84

Table 3. Relative improvement over the baseline performance [2].

On the other hand, Table 2 also shows that the approximant class exhibits the lowest classification rate (59.11%). This result may be explained by the presence of the element "hh" in this class, which is indeed closer to a fricative sound, as it can be understood looking at the spectrogram given in Figure 3.1.

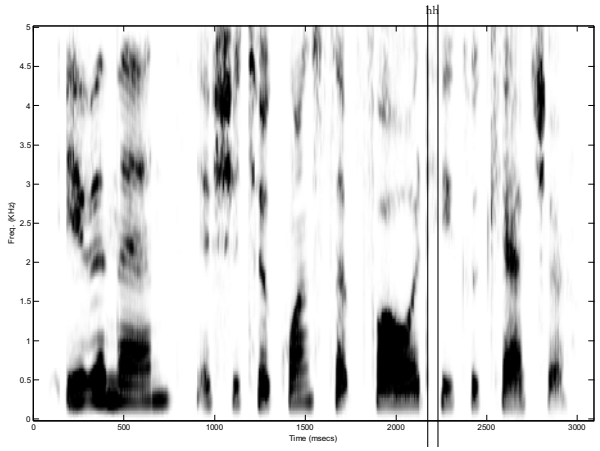

Figure 3.1. Spectrogram of the TIMIT sentence "Even then, if she took one step forward he could catch her." The selected area corresponds to the "hh" sounds.

To confirm this hypothesis, a further experiment was accomplished. In this experiment the "hh" element was eliminated from the approximant class, and the approximant detector was retrained, obtaining 7.44% performance improvement.

4. CONCLUSIONS

The aim of this paper was to show the importance of selection of an appropriate architecture for the detector component of knowledge-based ASR system. Experimental results have shown the effectiveness of "non-conventional" schemes. ANNs are a way to classify generic event, yet they are also a means to investigate the intrinsic nature of the data. Consequently the understanding of their mapping capability must be taken into consideration when used as technique to implement a detector or a classifier module. The results obtained in this paper are encouraging to further investigate the nature of non-monotonic hyperspherical neurons, and ANNs with learning activation functions.

5. ACKNOWLEDGMENTS

The first author is indebted with Prof. Chin.-H Lee and Adriane S. Durey for the insightful discussions on the topics and with Jinuy Li for his help in defining the general experimental framework.

6. REFERENCES

[1] K. Kirchhoff, "Combining articulatory and acoustic information for speech recognition in noisy reverberant environments," *Proc. ICSLP89*, Sydney, Australia, 1998.

[2] J. Li, Y. Tsao, and C.-H. Lee, "A study on knowledge source integration for rescoring in automatic speech recognition," *Proc. ICASSP05,* Philadelphia, 2005.

[3] B. Launary, O. Siohan, A.C. Surendran, and C.-H. Lee, "Towards knowledge-based features for HMM based large vocabulary automatic speech recognition," *Proc. ICASSP02*, Orlando, pp. 817-820, 2002.

[4] K. Hacioglu, B. Pellom, and W. Ward, "Parsing speech into articulatory events," *Proc. ICASSP04*, Montreal, Canada, pp.925-928, 2004.

[5] A. Juneja and C. Espy-Wilson, "Segmentation of continuous speech using acoustic-phonetic parameters and statistical learning," *Proc. International Conference on Neural Information Processing*, Singapore, 2002.

[6] J. Li, and C.-H. Lee, "On designing and evaluating speech events detectors," *Proc. InterSpeech05*, Lisbon, Portugal, 2005.

[7] S. M. Kay, *Fundamentals of Statistical Signal Processing Volume II: Detection Theory*, Prentice Hall, 1998.

[8] M. Gioiello, F. Sorbello, G. Vassallo, and S. Vitabile, "A VLSI neural device with sinusoidal activation function for handwritten classification," *Proc. Of 8th Conference on Neural Networks and their Applications*, pp. 238-242, 1996.

[9] Vladimir Banarer, Christian Perwass, and Gerald Sommer, "The hypersphere neuron," Proc. 11th ESANN, Belgium, 2003

[10] M. Gori, and A. Tesi, "On the problem of minima in backpropagation," *IEEE Trans. PAMI*, **14**, 1, pp. 76-85, 1992.

[11] D. G. Stork, and J. D. Allen, "How to solve the N-bit parity problem with two hidden units," *Neural Network*, **5**, pp. 923-926, 1992.

[12] M. R. W. Dawson, and D. P. Shopfocher, "Modifying the generalized delta rule to train networks of non-monotonic processor for pattern classification," *Connection Science*, **4**, 1, pp. 19-31, 1992.

[13] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discov.*, **2**, pp. 121-167, 1998.

[15] H. Li, D. Hestenes, and A. Rockwood, "Generalized homogeneous coordinates for computational geometry," *Geometric Computing with Clifford Algebra*, Spring-Verlag, pp. 27-52, 2001.

[16] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus," U.S. Dept. of Commerce, NIST, Gaithersburg, MD, February 1993.