Performance Comparison of Noise Classification Using Intelligent Networks

T.Meera Devi¹, N.Kasthuri² & A.M.Natarajan³
¹²Department of ECE, Kongu Engineering College, Erode
³Chief Executive Officer and Professor, Bannari Amman Institute of Technology, Sathyamangalam

Abstract: The performance of speech-processing systems such as speech coding, speech recognition has been degraded by background environmental noises such as car, bus, babble, factory, helicopter and street noise. Noise classification is essential to enhance the performance. A major step in the design of a signal classification system is the selection of a good set of features that are capable of separating the signals in the feature space. In order to reduce the effect of environmental noises on speech processing tasks, noise classification is required. In general, classification of noise module achieves an improvement in the performance of a system operating in the presence of background noise by dynamically adapting the processing algorithms to the particular type of environmental noise. In this proposed work fuzzy ARTMAP network and modified fuzzy ARTMAP network are used for classification of background noise signals. Further these results are compared with backpropagation networks and with Radial Basis Function Network(RBFN). Our empirical results show that the fuzzy networks have robust features in distinguishing the different classes of noises.

Keywords: Fuzzy ARTMAP, RBFN, Backpropagation Networks

1. INTRODUCTION

Speech enhancement in noisy environment is a challenging problem since decades. Noise get added in speech signal almost in an uncontrolled manner. Speech processing system encounters different types and levels of background acoustical noises. Systems like speech coding, speech recognition, speaker verification pick up those ‘unwanted’ signals along with speech. The noise signals result in performance degradation of those systems. For example, the accuracy of a speech recognition device might severely be affected if the level of noise is high and there is a mismatch between training and testing conditions. In speech coding, background noises can be coded with annoying artifacts.

Neural network has been considered for classification, as it has found success in the area of pattern classification problems. By repeatedly showing a neural network input classified into groups, the network can be trained to discern the criteria used to classify, and it can do so in a generalized manner allowing successful classification of new inputs under various environments. Many efforts to design automatic systems for controlling noise are done [1].This paper proposes the classification of background noise such as Factory noise, Babble noise and Car noise using networks namely Levenberg-Marquardt Backpropagation Network, Resilient Backpropagation Network and Bayesian Classifier, Radial Basis Function Network (RBFN), Fuzzy ARTMAP Neural Network (FAMNN) and Modified fuzzy ARTMAP Neural Network (MFAMNN).

Section 2 of this paper focuses on back propagation networks module. Section 3 confers the RBFN classification algorithm. In Section 4 fuzzy ARTMAP technique for noise classification is described. More specifically in Section 5 the importance of modified fuzzy ARTMAP algorithm is proposed and developed to implement the noise classification. Classification results from different tests of the classification algorithms are presented in Section 6. Finally, the conclusions of this paper are discussed in Section 7.

2. BACK PROPOGAATION NETWORKS FOR NOISE CLASSIFICATION

A major step in the design of a signal classification system is the selection of a “good” set of features that are capable of separating the signals in the feature space. The classifier operates on a frame-by-frame basis using short segments of the signal, e.g. 20 ms [2]. Various features are Zero crossing rate, Linear predictive analysis, Line Spectral Frequencies, Perceptual linear prediction (PLP), Cepstral analysis, Root-mean-square energy, Critical bands energies, Correlation coefficients etc.

Feature extraction is needed to reduce the dimensionality of the data passed to the neural network. The neural network will determine the best way to process the data to arrive at a classification. Extracting useful features from a digital audio sample is an evolving science and remains a popular research field [5]. Some of the features considered for classifications are Correlation coefficients, Cepstral coefficients and Line Spectral Frequencies.
In this paper, Networks considered for comparison of noise classification with fuzzy networks are Levenberg-Marquardt algorithm, Resilient backpropagation algorithm and Bayesian Classifier.

Speech-processing tasks need to discriminate speech from noise. These noise patterns have been classified using Levenberg-Marquardt algorithm, Resilient back propagation algorithm and Bayesian classifier. Here three commonly encountered noise types have been considered: factory, babble and car. A total of 30,000 frames (0.033ms), equally distributed between the three classes, were used for training.

Table 2.1 gives a brief summary on various network parameters for Levenberg-Marquardt, Bayes Classifier, Resilient backpropagation networks.

<table>
<thead>
<tr>
<th>Networks</th>
<th>Levenberg Marquardt</th>
<th>Bayes Classifier</th>
<th>Resilient Back Propagation</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of Epochs</td>
<td>25</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>Trained Time (sec)</td>
<td>6.15</td>
<td>365.53</td>
<td>31.91</td>
</tr>
<tr>
<td>Learning Rate ((\eta))</td>
<td>0.02</td>
<td>0.1</td>
<td>0.02</td>
</tr>
<tr>
<td>No. of Neurons</td>
<td>100</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>Transfer Function</td>
<td>Tansig &amp; Linear</td>
<td>Tansig &amp; Linear</td>
<td>Tansig &amp; Linear</td>
</tr>
</tbody>
</table>

Noise classification using back propagation networks like Levenberg-Marquardt algorithm, Resilient backpropagation algorithm and Bayesian classifier has been simulated using MATLAB 7.5.

3. RBFN-CLASSIFIER

RBFN provides a powerful alternative to Multi Layer Perceptron (MLP) neural networks to approximate or to classify a pattern set. This is becoming an increasingly popular neural network with diverse applications. Much of the inspiration for RBF networks has come from traditional statistical pattern classification techniques.

The spread value of the Gaussian function plays a vital role in the classification problem. The classification in RBF networks has been more accurate when the spread value is made nearer to zero. This means the network acts as a nearest neighbor classifier.

3.1. Network Architecture

RBF network has three layers as shown in Figure 3.1.

3.1.1. Input Layer

The input layer is simply a fan-out layer and does no processing. It serves as the source of nodes that connects neural network with the environment.

3.1.2. Hidden Layer

Each neuron in this layer consists of a radial basis function centered on a point with as many dimensions as there are predictor variables. The spread value of the RBF function may be different for each dimension. The centers and spreads are determined by the training process. When presented with the \(x\) vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron’s center point and then applies the RBF kernel function to this distance using the spread values. The resulting value is passed to the summation layer. The second or hidden layer performs a non-linear mapping from the input space into a higher dimensional space in which the patterns become linearly separable.

3.1.3. Output Layer

The final layer performs a simple weighted sum with a linear output. The output layer applies a linear transformation from the hidden space to output space. The value from the hidden layer is multiplied by the weights associated with the neuron and passed to the summation. For classification problems, there is one output (and a separate set of weights and summation unit) for each target category.

4. FUZZY-NETWORK

The fuzzy pattern classification of environmental noise presents: 1) a criterion to group a large range of environmental noise into a reduced set of classes of noise with similar acoustic characteristics; 2) a larger set of background noise together with a new multilevel classification architecture; 3) a new set of robust acoustic parameters; 4) a robust, computationally simple background noise fuzzy classifier based on the selected acoustic features. Noise classification is one of the major tasks of speech recognition and noise cancellation systems.
I. Introduction

The coding layers $F_0$ which generates the vector $A = (a, a')$ in ART and $B = (b, b')$ in ART. For reasons of simplification of the writings, let us note $I$ vector $A$ or $B$ according to whether it is about the vector of input of ART, or ART.

- The vector $X$ ($X_0$ for ART and $X'$ for ART) expresses the activation of F1.
- The vector of the adaptive weights binding $F_i$ and $F_j$ is noted ($W_i^a$ for ART and $W_j^b$ for ART). The vector $y$ ($y_0$ for ART and $y'$ for ART) expresses the activation of $F_j$.

The fuzzy ARTMAP has in addition to the three parameters of each fuzzy ART, three other parameters which are: The minimum value of the parameter of vigilance of ART noted $p^*$, the vigilance parameter $p^a$ and the training parameter $p^b$ of layer MAP.

Preprocessing of the input patterns takes place before they are presented to the ART module. The first preprocessing stage takes as an input an $M$-dimensional input pattern and transforms it into a vector $a = (a_1, ..., a_M)$ whose every component lies in the interval [0,1]. The second preprocessing stage accepts the vector as an input and produces a vector $I$, such that

$$I = (a, a') = (a_1, ..., a_M, a_1^c, ..., a_M^c)$$ (4.1)

Where

$$a_i^c = 1 - a_i, 1 \leq i \leq M$$

II. Fuzzy ARTMAP Neural Network

4.1. Fuzzy ARTMAP Neural Network

Traditionally, Fuzzy systems and neural networks have been investigated along rather different lines due partly to the fact that they are derived from rather different fields. While a fuzzy system possesses great power in representing linguistic and structured knowledge by fuzzy sets and performing fuzzy reasoning by fuzzy logic in a qualitative manner, it usually relies on domain experts to provide the necessary knowledge for a specific problem.

It is quite natural to consider the possibilities of integrating the two paradigms into a new kind of system where the desired strengths of both systems are utilized and combined with an expectation that the hybrid system will be greatly enhanced.

The fuzzy ARTMAP network shown in figure 4.2 is a supervised training neural network i.e. the training is controlled by a base of examples, where each example is an association of an input vector to a desired output vector. Its architecture is evolutionary, and it is composed of two fuzzy ART networks ART and ART. These two networks are bound by a network of a neural cells MAP. ART receives the bodies of the vectors of input and ART receives the associated vector of desired output. The inter-ART module determines whether the mapping between the input and the output is the correct one.

In the case where the mapping between inputs and outputs is many to one, the FAMNN operations can be completely described by referring only to the ART module. Each fuzzy ART module has three layers:

- The coding layers $F_0$ which generates the vector $A = (a, a')$ in ART and $B = (b, b')$ in ART. For reasons of simplification of the writings, let us note $I$ vector $A$ or $B$ according to whether it is about the vector of input of ART, or ART.
- The vector $X$ ($X_0$ for ART and $X'$ for ART) expresses the activation of F1.
- The vector of the adaptive weights binding $F_i$ and $F_j$ is noted ($W_i^a$ for ART and $W_j^b$ for ART). The vector $y$ ($y_0$ for ART and $y'$ for ART) expresses the activation of $F_j$.

The fuzzy ARTMAP has in addition to the three parameters of each fuzzy ART, three other parameters which are: The minimum value of the parameter of vigilance of ART noted $p^*$, the vigilance parameter $p^a$ and the training parameter $p^b$ of layer MAP.

Preprocessing of the input patterns takes place before they are presented to the ART module. The first preprocessing stage takes as an input an $M$-dimensional input pattern and transforms it into a vector $a = (a_1, ..., a_M)$ whose every component lies in the interval [0,1]. The second preprocessing stage accepts the vector as an input and produces a vector $I$, such that

$$I = (a, a') = (a_1, ..., a_M, a_1^c, ..., a_M^c)$$ (4.1)

Where

$$a_i^c = 1 - a_i, 1 \leq i \leq M$$

4.2. Training Phase

Given a list of MP, such as $\{F, O^1\}, \{F, O^2\}, ..., \{M^p, O^{mp}\}$ to train the FAMNN and to map every input pattern of the training list to its corresponding label. In order to achieve this goal, the training set is presented repeatedly to the FAMNN until the desired mapping is established for all pairs.

Consider the input/label pair (for example, $\{F, O^1\}$) from the training list. The bottom-up inputs to all the nodes at the $F^a$ field of the ART module due to the presentation of the input pattern are calculated. These bottom-up inputs to a node $j$ in $F^a$ are calculated according to the following:

$$T_{j}^{a}(I) = \frac{|I^a \wedge W_{j}^{a}|}{\beta_a + W_{j}^{a}}$$ (4.2)

Where $\beta_a$ is called the ART choice parameter, and takes values in the interval (0, $\alpha$). From the set of nodes in $F^a$ that satisfy the vigilance criterion, is chosen the one that receives the maximum bottom-up input from $F^a$. A node satisfies the vigilance criterion if

$$|I^a \wedge W_{j}^{a}| \geq \rho^a$$ (4.3)
Where $\rho^a$ is called the vigilance parameter, and takes values in the interval [0, 1]. Each time an input pair is presented, it is initialized to a value called the baseline vigilance parameter $\rho^a$.

### 4.3. Testing Phase

The values of the top-down weights and the fuzzy ARTMAP parameter values $(\beta_a, \rho_a)$ are set to the values that they had at the end of the training phase. For each input pattern from the test list, the bottom-up inputs as they were defined by considering all committed nodes and an uncommitted node is calculated. From the set of nodes in $F_2^a$ that satisfy the vigilance criterion, the one that receives maximum bottom-up input is chosen and the label of the input pattern is designated as $\hat{O}$, where $\hat{O}$ is the label that the node has been mapped to, in the training phase. If an uncommitted node is chosen, the label of the input pattern is designated as "unknown."

$$\frac{|I^r \wedge W^a_j|}{|I^r|} \geq \rho^a$$

### 5. MODIFIED FUZZY ARTMAP

Modified–Fuzzy ARTMAP is for incremental supervised learning and multidimensional mapping using fuzzy signals. This neural architecture is able to process and store linguistic information in the form of fuzzy logic rules and membership functions for fuzzy inference. The fuzzy ARTMAP network has too many parameters to be fixed to reach a rate of reasonable training. These parameters are: vigilance coefficient and training coefficient of ART$_a$, ART$_b$ and the MAP, and the comparison coefficient of ART$_a$ and ART$_b$. The mapping field in Modified-FUZZY ARTMAP can perform a one-to-one, many-to-one or one-to-many mapping. The one-to-many mapping ability helps in the case of noisy and / or uncertain training data.

In Modified–Fuzzy ARTMAP Neural Network the testing phases is different from the Fuzzy ARTMAP Neural Network. In this Modified–Fuzzy ARTMAP Neural Network the patterns are found out for each signal and are given as the input signal. In pattern recognition approach the matching phase is performed using a set of fuzzy rules. The fuzzy system was automatically extracted. The parameters are,

$$\rho^a = 0.1, \beta^a = \beta^b = 1, \alpha^c = \alpha^d = 0$$

(5.1)

The set of feature vectors extracted is divided into training and test sets. The training phase of MFAMNN is exactly the same as the training phase of FAMNN. The test phase is a modification of the test phase of FAMNN that exhibits superior generalization performance than the standard FAMNN if the textures are affected by more noise. The training phase is implemented in a way so that the modification does not introduce any extra overhead to the test phase. Two feature sets are used in this work, namely energy-based and fractal-based features.

### 5.1. Training and Test Phase

The training of the FAMNN-m is exactly the same as the training phase of the standard FAMNN. The bottom-up input of node $j$ in $F_2^a$ for a pattern $I$, denoted as $T^a_m(I)$, can be expressed as a ratio of two numbers $N_j$ and $D_j$. In particular

$$D_j = \beta^a + |W^a_j|$$

(5.2)

$$N_j = |I^r \wedge W^a_j|$$

After the training is over, $D_j$ remains unchanged. Even if the training is on-line, this quantity remains unchanged when the test phase takes place. The quantities $D_j$ for every node $j$, are stored in memory along with the templates $W_j$, so that they are not recalculated in the test phase.

Computing time is more important while training a network. The solution in making to learn the modified fuzzy ARTMAP offers the best compromise classification quality or computing time. For the fuzzy ARTMAP with search of optimal parameters this time is of $(1+1/\lambda) 8$ time of training time of a traditional ARTMAP. But in the same application, for the modified ARTMAP, this time is proportional to the number of passage which is multiplied by the training time of a traditional ARTMAP.

ARTMAP network, Fuzzy ARTMAP network and modified fuzzy ARTMAP are simulated using MATLAB. The results and comparison between the three networks are shown in table 6.2.

### 6. SIMULATION RESULTS

Simulation results for back propagation networks, Radial basis function network fuzzy ARTMAP network and modified fuzzy ARTMAP network are shown in.

#### 6.1. Noise Waveforms

**6.1.1. Factory Noise**

![Figure 6.1: Factory Noise](image)
The Figure 6.1 shows the input waveform of background factory noise which is taken upto 30,000 samples. The figure 6.2 shows the input waveform of background babble noise samples and the figure 6.3 shows the input waveform of background car noise.

6.1.2 Babble Noise

Fuzzy networks for classification of noise signals are also simulated using MATLAB 7.5. The table 6.2 shows the performance comparison of ARTMAP network, fuzzy ARTMAP network and modified fuzzy ARTMAP network.

6.1.3. Car Noise

Table 6.3 provides a comparative study on various networks for background noise classification. The result shows that the noise classification using RBFN gives better performance than back propagation networks. When compared to the RBFN the proposed network provides the best performance in noise classification.

7. CONCLUSION

The classification of noises for various environmental backgrounds using features such as Line Spectral Frequencies, Cepstrum and Correlation coefficients has been simulated using MATLAB. Frame level noise classification have been performed using Back propagation networks,
Radial Basis Function network classifiers, Fuzzy ARTMAP and modified fuzzy ARTMAP network. The results of these networks are compared. The proposed modified fuzzy ARTMAP network improves the noise classification 99.6% over 87.33% in back propagation networks and 94% in Radial Basis Function Network with faster convergence.

The observations include

- MFAMNN are more efficient than RBFN;
- RBF Networks are more efficient than back propagation networks;
- Training time of MFAFNN is lesser than compared to other networks.

Considering the overall classification accuracy of these networks used for background noise classification, modified fuzzy ARTMAP classifier outperforms the other classifiers tested.

The results of noise classification can be used to cancel out the predominant noises. This needs modeling of various environmental noises. The modeled noise can be used as a reference for the noise cancellation algorithm and the classified noise can be cancelled out. Also this noise classification can be applied for speech recognition in mobile environments.

REFERENCES


