Neural Networks vs Genetically Optimized Neural Networks in Time Series Prediction

Ion Railean[1,2] Sorin Moga[1] Monica Borda[2] Cristina Stolojescu[1,3]

[1] Institut TELECOM; TELECOM Bretagne, UMR CNRS 3192
Lab-STICC; Université européenne de Bretagne, France
(e-mail: `firstname.lastname@telecom-bretagne.eu`)
[2] Technical University of Cluj-Napoca, Romania
Faculty of Electronics, Telecommunications and Information Technology
(e-mail: `firstname.lastname@com.utcluj.ro`)
[3] Politehnica University of Timisoara, Romania
Faculty of Electronics and Telecommunications
(e-mail: `firstname.lastname@etc.upt.ro`)

Abstract. This paper deals with methods for finding the suitable weights in an Artificial Neural Network (ANN) using Genetic Algorithms (GA). We study the weakness and strength-ness of the proposed approach in case of a statistical data forecasting. We describe a different approach when using the input data during optimization phase. Besides GA, we applied stationary wavelet transform (SWT) as a signal preprocessing, and time-delay neural networks (TDNN) approach for the system's inputs. Our results show that this optimization is suitable only for certain purposes in case of a statistical data prediction.
Keywords: Genetic Algorithms, Artificial Neural Networks, forecasting.

1 Introduction

The optimization of Artificial Neural Networks using Genetic Algorithms applied in forecasting have been proposed in many papers [1], [5], [6]. In [1] is presented the way of determining the optimal size of the hidden layer and the number of connections between layers. In [2] an approach using genetic computing is given, used for establishment of the optimum number of layers and the number of neurons on layer, for a given problem. A proposal of an intelligent algorithm to select the optimal architecture for ANN model in hot rolling process based on GA is shown in [3]. Venkatesan [4] proves the importance of the accuracy of algorithm-based ANN model for the turning process in manufacturing industry. The simultaneous optimization of the network architecture and the training of weights is presented in [7]. Most of the papers present the use of optimized ANNs in forecasting only in industrial processes, which are described by well-predefined formulas and the selection of parameters is required, and do not depend on statistical and human behavior.

In this paper we try to understand the influence of the ANN optimization using GA in a domain implying statistical data: WiMAX network traffic. We make a comparison between prediction accuracy of the optimized and un-optimized ANNs. Our optimization consists in setting the weights of the neural networks. In comparison to other researchers, we propose a new approach in selection of the training data.

Another aspect, is that we use wavelet transform as a signal preprocessing, and the ANN optimization is done for each of the signal's decomposition level.

The rest of the paper is organized as follows. The sections 2, 3, and 4 describe the basic aspects of the GA, ANN, and SWT. The section 5 shows the simplified forecasting framework used in our analysis. The experiments, results, and the comparison between regular ANN training and optimized ANN are given in section 6. While section 7 contains the main conclusions of the current research.

2 Genetic Algorithms

A GA is a search technique for optimization and machine learning applications. It is based on natural selection, the process that drives biological evolution. It consists of a set of individual elements (the population). At each step, the GA selects individuals randomly from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution. There are several steps in a GA:

- Encoding technique: gene, chromosome
- Initialization procedure: creation
- Evaluation function: environment
- Selection of parents: reproduction
- Genetic operators: mutation, recombination
- Parameter settings: practice and art

The population members are strings or chromosomes. The GA selects a subset (usually pairs) of solutions from a population, called parents, and combines them to produce new solutions called children or offsprings. The rules of combination to yield children are based on the genetic notion of crossover, which consists of interchanging solution values of particular variables. There are also occasional operations such as random value changes, which are called mutations. The children produced by the mating of parents, and that pass a survivability test, are then available to be chosen as parents for the next generation.

3 Artificial Neural Networks

The ANN is a mathematical model that simulates the structure and functions of the real biological neural networks. It is composed by interconnected simple elements, called artificial neurons. An ANN is characterized by three things:

- Its architecture: the pattern of nodes and connections between them
- Its learning algorithm, or training method: the method for determining the weights of the connections
- Its activation function: the function that produces an output based on the input values received by a node

The two most important types of ANNs are feed-forward (FFANN) and recurrent networks (RANN). A FFANN has its neurons organized in a layered structure. Each layer consists of units which receive their input from the units situated on a layer directly below and send their output to the units from a layer directly above. RANN are characterized by the fact that they contain feedback connections and they take into consideration the dynamical properties of the network.

In this paper we used feed-forward ANN, and we discuss the setting of weights of the connections. One way is to set them explicitly, using a priori knowledge. Another way is to "train" the ANN by feeding it teaching patterns and letting it change its weights according to some learning rule. While our approach, consists in applying GA to find the optimal weights between the input and the hidden layer.

4 The wavelet analysis

Multi-resolution analysis (MRA) is a signal processing technique that takes into account the signal's representation at multiple time resolutions. Using wavelet MRA, the collected measurements can be smoothed until the overall long-term trend is identified. Fluctuations around the obtained trend are further analyzed at multiple time scales. The level of decomposition depends on the length of the data set (the number of values). At each temporal resolution two categories of coefficients are obtained: approximation and detail coefficients. We used the à trous methodology in MRA implementation, also known as Shensa's algorithm [9], which corresponds to the computation of the Stationary Wavelet Transform (SWT).

The à trous wavelet transform decomposes a signal X_t as follows:

$$X_t = a_{p,t} + \sum_{j=1}^{p} d_{j,t} \tag{1}$$

where $a_{p,t}$ represents the smooth version of the original signal (the approximation at the p^{th} level of decomposition), while $d_1...d_p$ represent the details of X_t at scale 2^{-j}. This equation can be seen as a multiple linear regression model also, where the original signal is expressed in terms of its coefficients. Among different mother wavelets (Daubechies, Symlet, Meyer, etc. [8]), we used Daubechies 2 wavelet.

5 Forecasting framework

The simplified forecasting framework of our analysis is presented in Figure 1.It implies a series of steps as presented below:

1. Use SWT to decompose the data for input and for test
2. Apply an Artificial Neural Network for each level of decomposition obtained from the input and build the forecasting model. Choose between having or not the GA Optimization for the ANN
3. Use the decomposed test data and the obtained model in order to predict each decomposition level of the future forecasted signal
4. Use the Inverse SWT in order to obtain the final predicted signal

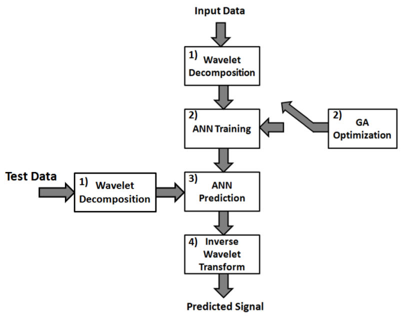

Fig. 1. The main block diagrams of the forecasting framework. Each block represents one of the steps taken in the construction of our model

6 Experiments and results

6.1 Data analysis

Our WiMAX traffic data used in analysis was obtained by monitoring the traffic from 67 Base Stations (BS) during eight weeks, from March 17th till May 11th, 2008. Each BS has its own data set which is composed of numerical values representing the total number of packets from the uplink channel. Each value is recorded every 15 minutes. It results that for a given BS we have the following number of samples: 96 samples/day, 672 samples/week, and a total number of 5376 samples.

6.2 ANN approach

The goal in our experiments was to make one day ahead forecasting. Taking into account this information we used for ANN's architecture only one neuron for the output, which consists of an array of 96, 32, or 16 samples. 32 and 16 samples were obtained by making a downsampling of the signal with 3 and 6. These downsamplings were done because of the existence of observed periodicities in the WiMAX traffic [13]. Regarding the number of layers, we used one hidden layer network. In [2] is pointed out the fact that one hidden layer network is able to approximate most of the nonlinear functions demanded by practice.

For the input layer, we used the approach of TDNN described in [10], [11], and [7]. The time-delay of the input information was set to 4, 8, 12, and 24 hours shifting. The data used for the input layer was the wavelet transform obtained from the weeks 1-6 during training process, and weeks 2-7 during test. Also, we did not apply all the data at ANN's inputs, we used only the days corresponding to the same period of the week as the forecasted day. The number of neurons for the hidden layer was 2. The training algorithm was the combination between adaptive learning rate with momentum.

6.3 Genetically Optimized ANNs

For the optimized neural networks we used an approach permitting us to train a given ANN using two data sets at the same time. In the first part we used the information from weeks 1-5 while having as a target the given day from week 6, and in the second part we used the information from weeks 2-6, while having as a target the information from week 7. During testing process, we applied at the optimized ANN's inputs the wavelet transform from corresponding to the days from weeks 3-7. An example of this training is given in Figure 2 (because of the space, we present in the figure a simplified optimization using only 2 days at the input). The final predicted signal, obtained after applying inverse stationary wavelet transform on all forecasted sequences, was compared to the original signal from 8th week.

The designing of training the ANN using the Genetic Algorithms is as follows:

- each individual contains a set of weights for all the links between layers
- each gene represents a single weight
- we had a population size of 100 individuals, meaning 100 different possibilities at each generation for the network
- the number of generations is 100: less generations resulted in not finding an acceptable solution for our problem, while more generations resulted in a longer time processing. However, above this value, we didn't manage to observe better performance of the final results
- the fitness function is calculated as follows:

$$F = \frac{1}{N}\sum_{i=1}^{N}(x_i^{f_1} - x_i^{o_1}) + \frac{1}{N}\sum_{i=1}^{N}(x_i^{f_2} - x_i^{o_2}),$$

  where N is the number of samples, $x_i^{o_1}$and $x_i^{o_2}$ represent certain level decomposition of the original signal used for inputs, while $x_i^{f_1}$ and $x_i^{f_2}$ are the output targeting signals for the two data sets.

6.4 Evaluation criteria

In order to evaluate the prediction performance between ANNs and genetically optimized ANNs, we used the following well-known evaluation criteria: Symmetrical Mean Absolute Percentage Error (SMAPE) and R-Square (RSQ):

- SMAPE: calculates the symmetric absolute error in percent between the actual X and the forecast F across all observations t of the test set of size n for each time series s:

$$SMAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|X_t - F_t|}{(X_t + F_t)/2} \tag{2}$$

- RSQ: the coefficient of determination R^2, in statistics, is the proportion of variability in a data set that is accounted for by a statistical model. In this definition, the term *variability* is defined as the sum of squares:

$$R^2 = \frac{SS_R}{SS_T} = \frac{\sum_t (X_t - \overline{X_t})^2}{\sum_t (F_t - \overline{F_t})^2} \tag{3}$$

in which X_t, F_t are the original data values and predicted values respectively, while $\overline{X_t}$ and $\overline{F_t}$ are the means of the observed data and modeled (predicted) values, respectively. SS_T is the total sum of squares, SS_R is the regression sum of squares. In the ideal case the value of RSQ is 1, while SMAPE must be 0.

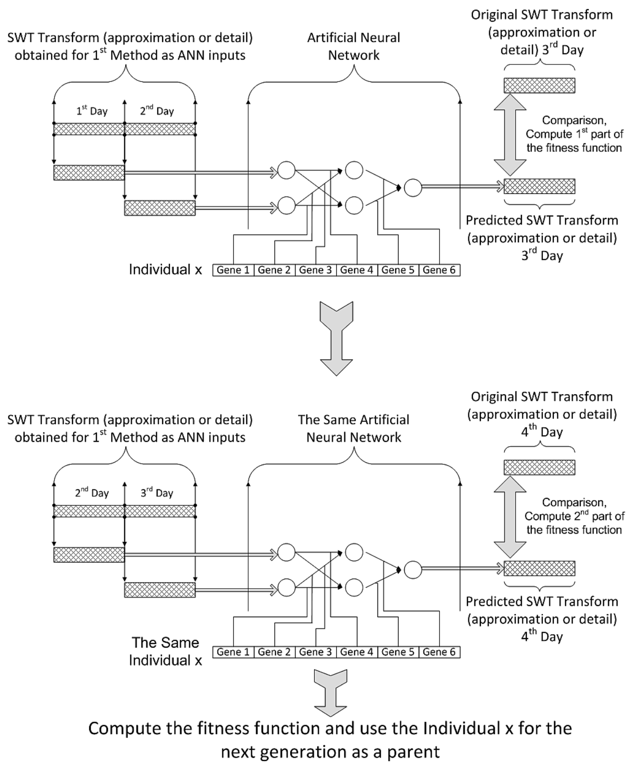

Fig. 2. Example of ANN Optimization. The top diagram represents the computation of the first part of the fitness function, while the bottom diagram exemplifies the computation of the second part of the fitness function

6.5 Results

In both ANN and genetically optimized ANNs we made 8232 simulations from 11256 possible (after extracting the erroneous data). We made a combination between all the possibilities in choosing the number of BS (from a total of 67), number of samples per day (16, 32, or 96), time-delay interval for inputs (4, 8, 12, or 24 hours shifting), and the day of the week (from Monday till Sunday). The results for RSQ and SMAPE values are presented in the tables 1 and 2. They present

Time Delay	Measured Value	96 samples	32 samples	16 samples
4 hours	RSQ	1.212	1.249	1.317
	SMAPE	0.905	0.927	0.802
8 hours	RSQ	**1.159**	**1.120**	**1.205**
	SMAPE	**0.886**	**0.858**	**0.756**
12 hours	RSQ	1.330	1.219	1.287
	SMAPE	0.904	0.924	0.814
24 hours	RSQ	1.178	1.201	1.263
	SMAPE	0.910	0.946	0.780

Table 1. RSQ and SMAPE values (one day prediction) using ANN. The best values are represented in bold

the medium value of the results from all 67 BS and days of the week during a given configuration of samples per day and the number of shifted hours.

By comparing the results, we can see that the SMAPE values in case of GA optimization are not better then the ones of the usual ANN training. However, the RSQ value is closer to the ideal 1. This is true for all used configurations of TDNN.

Time Delay	Measured Value	96 samples	32 samples	16 samples
4 hours	RSQ	1.127	1.200	1.176
	SMAPE	1.001	**0.913**	0.867
8 hours	RSQ	**1.108**	**1.129**	**1.099**
	SMAPE	**0.983**	0.924	**0.820**
12 hours	RSQ	1.211	1.155	1.168
	SMAPE	1.008	0.951	0.851
24 hours	RSQ	1.087	1.189	1.215
	SMAPE	1.067	0.950	0.872

Table 2. RSQ and SMAPE values (one day prediction) using Genetically Optimized ANN. The best values are represented in bold

7 Conclusions

In this paper we presented a comparison between neural networks and optimized neural networks used in statistical data forecasting. We proposed a combination between wavelet transform, time delay neural networks, and a different approach of using the input data when applying GA for our ANN training.

Our results show that in case of the optimization the SMAPE value increased with about 2-7% in comparison to the regular ANN training, while the RSQ value decreased with about 2-10%. It means that the optimized ANN is able to express better the variability of the statistical data. This is because it takes into consideration a longer time interval in optimization, as we managed to use two data sets at the same time during training process. However, the value of SMAPE is not better. The reason for these differences might be as follows: according to the RSQ formula, we make a ratio between the variabilities according to the medium values of predicted and real signal; while SMAPE presents the differences between predicted and real samples of the signal. This means that in case of optimized networks we have a more shifted medium value from the medium of the original signal in comparison to the regular ANNs, because it keeps the behavior of the data from twice longer time then in case of the other approach. While regular ANN, expresses the behavior closer to the data we want to predict, and it does not necessarily make a generalization of the earlier time intervals.

One of our future task is to test our approach on other non-statistical data sets. We will try also to integrate the obtained model of calculating more precisely the data variability in other methods of prediction. Also, we will use it especially in data classification, because we have obtained an approach that uses multiple data sets during network training, while in case of regular network training we would need a more complex architecture for the ANN in order to obtain the desired results.

Acknowledgements

We would like to thank Alcatel Lucent Timisoara who sent the data used in our research. We gratefully appreciate their effort and assistance.

References

1. R. Garcia Ojeda et al., *Genetic Algorithms in the Optimal Choice of Neural Networks for Signal Processing*, IEEE, 0-7803-2972-4, 1996
2. I. Ileana et al., *The Optimization of Feed Forward Neural Networks Structure using Genetic Algorithms*, ICTAMI, Thessaloniki, Greece, 2004
3. J. S. Son et al., *A study on genetic algorithm to select architecture of a optimal neural network in the hot rolling process*, Journal of Materials Processing Technology 153-154, pp. 643-648, 2004
4. D. Venkatesan et al., *A genetic algorithm-based artificial neural network model for the optimization of machining processes*, Neural Comput and Applic 18, pp. 135-140, 2009
5. J. C. Yu, Y. L. Tseng, *Evolutionary Engineering Optimization Using Recursive Regional Neural Network and Genetic Algorithm*, IEEE, 0-7695-2882-1/07, 2007
6. A. Fiszelew et. al., *Finding Optimal Neural Network Architecture Using Genetic Algorithms*, Research in Computing Science 27, pp. 15-24, 2007
7. B. T. Zhang, H. Muhlenbein, *Evolving Optimal Neural Networks Using Genetic Algorithms with Occam's Razor*, Complex systems 7, pp. 199-228, 1993
8. S. Mallat, *A Wavelet Tour of Signal Processing*, Second Edition, (1999)
9. M.J.Shensa, *Discrete Wavelet Transform. Wedding the a trous and Mallat algorithms,* IEEE Transactions and Signal Processing, 40, pp. 2464-2482,(1992)
10. T. Taskaya-Temizel et al., *Configuration of Neural Networks for the Analysis of Seasonal Time Series*, ICAPR 3, Lecture Notes in Computer Science, vol. 1, pp. 297-304, 2005
11. D. S. Clouse et al., *Time-Delay Neural Networks: Representation and Induction of Finite-State Machines*, IEEE Transaction on Neural Networks, vol. 8, no. 5, September 1997
12. G. P. Zhang, M. Qi, *Neural network forecasting for seasonal and trend time series*, European Journal of Operational Research 160, pp. 501-514, 2005
13. C. Stolojescu et al., *Forecasting WiMAX BS Traffic by Statistical Processing in the Wavelet Domain*, in Proceedings of ISSCC, Iasi, Romania, pp. 177-183, 2009