Chapter Contribution

Rodney M. Goodman
California Institute of Technology

Barry Ambrose
California Institute of Technology

22 February 1994
Version 1

Chapter 1

Neural Network Applications to Telecommunications and Network Management

1.1 Introduction

Multi-Layer artificial neural networks have traditionally been used for pattern recognition and pattern classification and are well suited to this task. The weights in such networks can be learned from sample data and training can be done using the well known backpropagation algorithm [1]. In this chapter, a survey is carried out of the application of such networks to various activities associated with telecommunications.

The neural network approach is valuable for a number of reasons. Firstly the neural network has the ability to learn. Many software systems in the telephone network require the intervention of humans if the implemented function is to be modified. An example may be ATM Call Admission.

If the software is setup to admit a fixed set of traffic classes, it requires reprogramming if it is faced with new traffic class. This reprogramming can require a lot of effort. Neural networks on the other hand contain their "program" in their weight settings and can even continuously update their weights to learn a new call acceptance problem as they are running. This automated learning capability is a key benefit of neural networks.

The ability to learn is tightly linked to the ability to generalize on data that was not present in the training set. Neural networks have been characterized as universal function approximators [2, 3, 4]. This explains the ability of neural networks to generalize well, even on input data that contains a lot of noise and artifacts.

The second benefit of using neural networks is the parallelism in the architecture of the networks. Each neuron in the network is a simple processing device that can be easily implemented in silicon. It can then provide its mapping from input to output almost instantly. The availability and utilization of dedicated neural network chips will provide new opportunities for high speed processing of data.

Present day telecommunications networks could already be characterized as massively parallel systems. Neural networks provide the capability to extend this parallelism to the circuit level.

The chapter is organized as follows. First an introduction is given to the different types of neural network. The remainder of the chapter is then arranged by application area. For each area, a report is given on current status of neural networks. The following areas are included:

- Network Routing

- ATM Admission Control
- Equalization and Filters
- Speech Recognition
- Time Series Prediction
- Support Activities

1.2 Neural Network Architectures

Four architectures are most commonly found in the neural network literature [1]. These are (i) perceptrons (ii) feed-forward neural networks, also known as multi-layer perceptrons (MLP) (iii) recurrent neural networks and (iv) Hopfield networks. An example of each is given in this section along with a short description of their similarities and differences. A good reference for all these architectures and associated training techniques is [1].

1.2.1 Perceptrons

A perceptron is the simplest type of neural network, whose output is a linear combination of its inputs. An example of a simple perceptron is shown in Figure 1.1. It can be trained to adjust its weights to yield a decision function that is sufficient for classification of problems that are linearly separable. A problem is linearly separable if all the positive examples are on one side of a hyperplane in its input space. The exclusive or function (XOR) is an example of a decision function that is not linearly separable and which cannot be learned perfectly by a perceptron.

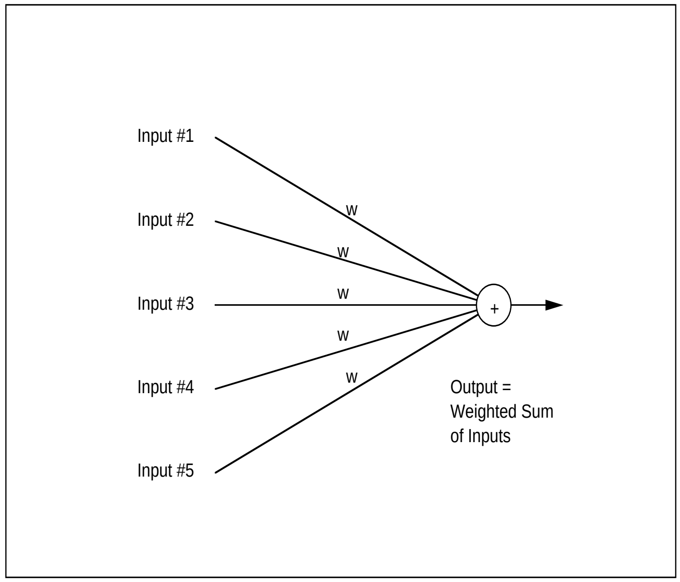

Figure 1.1: simple perceptron

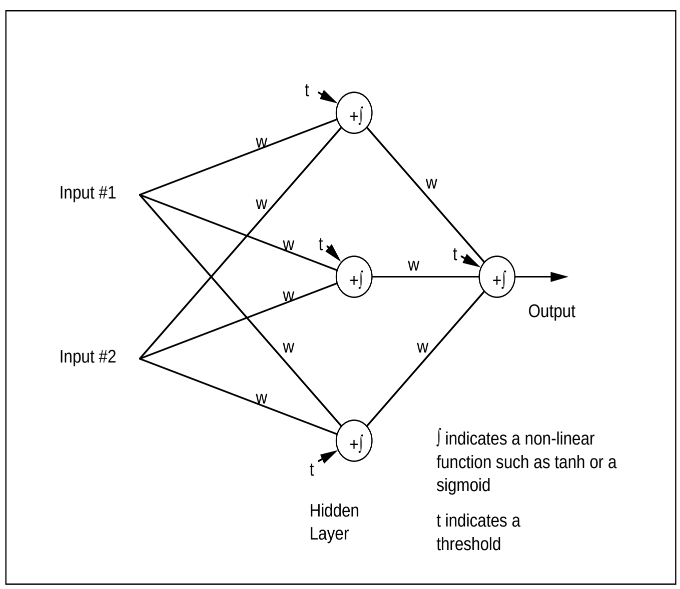

Figure 1.2: simple feed-forward neural network

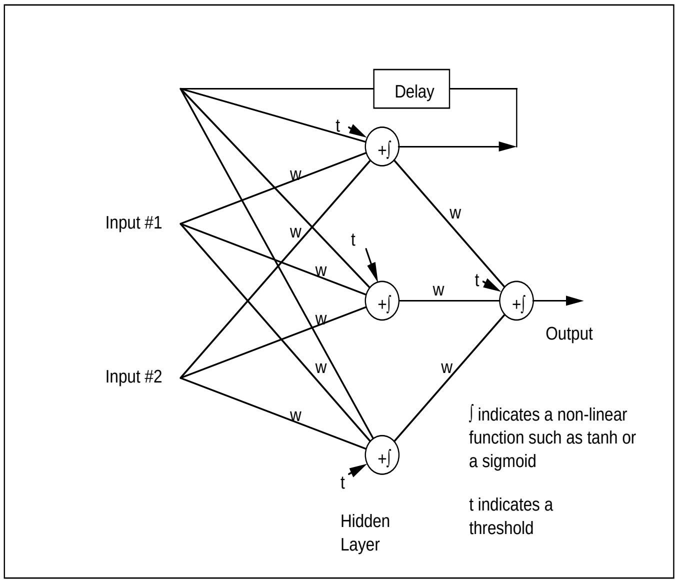

Figure 1.3: simple recurrent neural network

1.2.2 Feed-Forward Neural Networks

The feed-forward neural network is the most widely used neural network. An example of a simple feed-forward neural network is shown in Figure 1.2. It is ideal for pattern classification problems. The inputs are the measurements of the attributes of the object under study and the output is a binary indicator of which of two classes it belongs to. Of course there may be more that a single output, if there are more than two classes. The XOR problem *can* be learned by a feed-forward neural network.

1.2.3 Recurrent Networks

Recurrent Networks are similar to feed-forward neural networks except that feedback connections are also allowed. An example of a simple recurrent

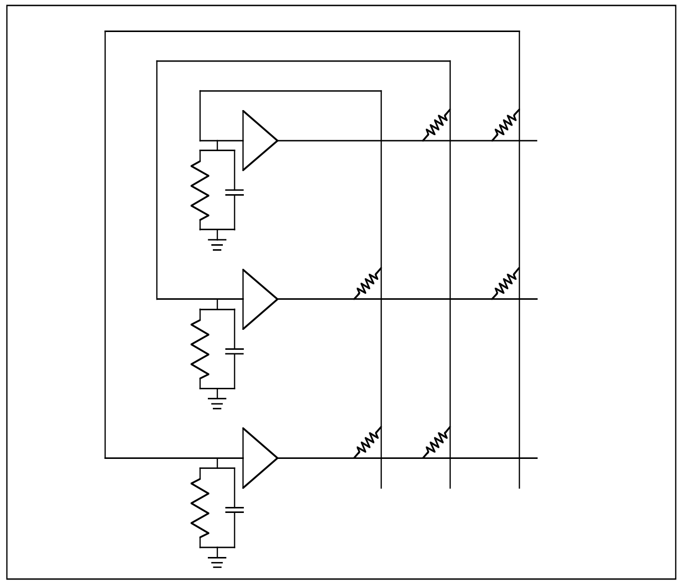

Figure 1.4: simple Hopfield neural network

neural network is shown in Figure 1.3. By using feedback connections, the state of the network at the present time is not only dictated by the present inputs but also by the inputs in the previous time-steps which are fed back. The recurrent networks can be likened to finite state machines and provide mappings between input and output that are similar. Training for recurrent networks is more complicated than that for feed-forward networks.

1.2.4 Hopfield Networks

Hopfield Networks are best described by the differential equations that govern their dynamics. They are comprised of amplifiers, resistors and small capacitors, all of which can all be implemented on silicon. An example of a simple Hopfield Network is shown in Figure 1.4. The mode of operation of the network is that a switch is thrown allowing currents to flow in the network and the network reaches an equilibrium state that corresponds to the solution of an optimization problem. The fact that it is a solution can be seen by analyzing the dynamics of the network and showing the "energy function" of the network decreases with time. The energy function implicitly encodes the optimization problem to be solved. Hopfield Networks have been successfully applied to the Traveling Salesman Problem, for example, where a salesman has to visit each city in a given list of cities once and minimize the total distance traveled. The weights of the network are hand-chosen to represent the problem constraints.

1.3 Network Routing

Network routing is by its very nature a distributed operation requiring each node in a network to make a decision on the best of many outgoing links available to route traffic based on a knowledge of its final destination and in some cases a knowledge of congestion situations elsewhere in the network.

Littman and Boyan in [5] describe an algorithm for packet routing in which learning is employed. The aim of the algorithm is to adjust the routing tables to achieve minimal routing times. The ability to learn routing rules shows promise for real networks where links can be added or taken away without the need for planning studies on the implications for routing policy. Also if links fail, the ability to learn new routing options should result in an improved grade of service. Experiments on a 36-node irregularly connected network showed the learning approach to be superior to routing based on the precomputed shortest path.

Distributed reinforcement learning is used. Each node keeps an estimate for each destination of how long it will take to reach the destination if it is sent on each of the available outgoing links. The outgoing link with the lowest estimated delivery time is chosen. Instead of then waiting for the packet to finally reach its destination before updating the policy, the node queries the winning neighbor for its estimated delivery time and uses that to update its own estimate, by factoring in its own delay. This policy relies on nodes nearer the destination to have better estimates of delay time. Simulation showed the algorithm performed equivalently to a fixed path algorithm for low loads, and better for high loads. For high loads, the learning algorithm was able to make use of longer paths that provided shorter delays under loaded conditions.

A slightly different approach is taken by Jensen, Eshera and Barash in [6]. Again they are concerned with routing tables in a packet switch network. Again dynamic update of the routing tables is carried out. Again simulation is used to compare the results with fixed routing tables.

The algorithm employed seeks to minimize time taken for a packet to travel from source to destination. The assumption is made that the time taken for a packet to travel from source to destination is identical to the time taken for a packet to travel from destination to source. This is called backward learning. The sending timestamp on each received packet is used to compute the travel time, which in turn is used to update the weight of that route to that destination. Simulations showed the algorithm performed equivalently to a fixed path algorithm for low loads.

Further references to the use of neural networks in dynamic routing are given in [7, 9, 10] which the interested reader can followup.

1.4 ATM Admission Control

In an ATM network, since resources are shared, the peak bit rate, mean bit rate and burstiness of a traffic source are important parameters. At the time the call is set up, the user may specify these parameter via a request for a communications channel with a given class of service. The network at call setup time must make a decision on whether to admit a call based on its declared class of service and based on some accepted quality of service criterion. In the statistically multiplexed environment, ATM admission control is an important function.

Hiramatsu and Takahashi in [8] propose an adaptable threshold for accepting new call setup requests using a neural network as a controller. Simulations show that the neural network can learn the correct threshold for allowed mixes of traffic that can be accepted. In the simulation the neural network had as inputs the number of calls currently in progress from each of two classes of traffic and the output was an indication of whether to accept or reject a new call. The network was trained by using cell loss rate as an error measure during the training. It was demonstrated for the two classes of traffic, that the network was able to learn the correct "accept or reject" boundary to achieve a given quality of service as specified by the acceptable cell loss rate.

Neves, de Almeida and Leitão in [32] also describe a scheme for using neural networks for ATM call control. The main difference is that the neural network is trained to give as outputs the expected delay, cell loss rate and maximum and minimum buffer occupancies from a given traffic mix. This allows acceptance or rejection of a new call based on the required quality of service objectives. A simulation of three service classes using a feed-forward neural network with 7 hidden units showed the neural network to work very well at achieving a good mix of traffic while keeping within the quality of service bounds.

Further references to admission control using neural networks can be found in [27, 29, 31] for the interested reader.

1.5 Equalization and Filters

Since feed-forward neural networks implement functions that are similar to linear filters, it is not surprising that many attempts have been made to replace linear filters with neural networks. The advantage of using a neural network is that a well known training algorithm, backpropagation, can be used to optimize the weights of the neural network with respect to minimizing a mean squared error function. Thus a feed-forward neural network can be regarded as a filter whose output is a non-linear function of its previous inputs.

Brown in [22] comments that some of the results for improvements due to neural networks should be interpreted with caution. In the paper, results previously obtained are reviewed and it is noted that while a 2-input feed-forward neural network is much better than a 2-input linear filter at channel equalization on a non-minimum phase channel, this advantage disappears as more tapped delay inputs are considered. Brown also shows that the use of direct connections from input to output can speed-up the training of neural network filters.

Cid-Sueiro and Figueiras-Vidal in [23] describe how recurrent radial basis function networks (RRBF) can be used to improve the performance over the standard Finite Impulse Response (FIR) approach. The assumptions made for the equalization problem are that the transmitted sequence x_k has been corrupted by possibly non-linear distortion and additive Gaussian White Noise. They show that the formula for the optimum decision on a symbol can be written in a form that is identical to that of a recurrent neural network that uses Radial Basis Function instead of sigmoids. A radial basis function is a function of the form

$$y = \exp(\frac{x^2}{r^2})$$

The parameters for the network are obtained from a knowledge of the channels impulse response.

Further references to channel equalization using neural networks can be found in [24, 25, 26] for the interested reader.

1.6 Speech Recognition

Speech recognition is an enabling technology that would allow more automated interactions between the customers and the network. The central problem is one of pattern recognition. Since neural networks are adept at pattern recognition, many successful attempts have been made at applying neural networks to the problem. The conventional approach is to use Hidden Markov Models or Dynamic Time Warping followed by Template Matching [12].

Kwasny, Kalman, Engebretson and Wu in [11] describe a network which is capable of classifying raw speech waveforms as being either English or French. They use a recurrent neural network which has been trained on examples of English and French. The waveform is divided into a number of segments and each segment is classified using the network. These classifications can be regarded as votes for a particular decision. At the end of the waveform, the votes are summed to yield a decision on the language of the speaker.

The inputs for their model were sampled frequency bands over time. Two hundred and forty inputs represented five frequency bands over 1.2 seconds. There were four hidden units used, with the hidden unit outputs used as inputs for the the next time step. The network classified both training and test sets correctly. For training to succeed, singular value decomposition was applied to the inputs patterns, re-orienting the data to maximize orthogonality among the input activations.

Lerrink and Jabri in [13] describe experiments with three different learning algorithms for the problem of phoneme recognition from spectral input data. Two of the methods implement partially supervised learning, that is, the correct classification is not provided and instead the network gets a Boolean indication of right or wrong for each classification it makes during training. The Boolean indication is given at word boundaries. The "temporal difference" (TD) algorithm is one of the most widely used in reinforcement learning research. The infinite input duration (IID) algorithm can be used to train a recurrent neural network directly. Finally fully supervised learning using Backpropagation Through Time (BTT) was used to train a recurrent neural network as a reference. The results for percentage correct classification in Table 1.1 indicate the difficulty of implementing learning without full supervision.

Further references to speech recognition using neural networks can be found in [28, 14, 15] for the interested reader.

Algorithm	Results (Percent correct)
TD	69%
IID	73%
BTT	76%

Table 1.1: Performance for Phoneme Recognition

1.7 Time Series Prediction

In network management, the ability to predict future traffic arrivals, occupancy levels, and error events on links can be a valuable tool. This can be done using time series prediction of the statistic of interest. The ability of neural networks to learn non-linear mappings means that the limitations of the standard Linear Predictor can be overcome.

Fishwick, Almeida and Tang in [16] carried out a study in which they compared the Box-Jenkins methodology [17] to using feed-forward neural networks. From their results they conclude both methods yield comparable results. The neural network models were seen to be robust and provided good long-term forecasts. They were also seen to be parsimonious in their data requirements.

Goodman and Ambrose in [18] describe how feed-forward neural networks can do slightly better than Linear Predictors (or perceptrons) for the prediction of telephone traffic occupancy on a trunk group. A six input network is used, with four hidden units. The inputs are a tapped delay line representation of the previous 6 observations of occupancy.

Further references to time series prediction using neural networks can be found in [19, 20, 21] for the interested reader.

1.8 Support Activities

There are many telecommunications related activities that are essential to the smooth operation of a telephone company. Examples might be billing of customers or negotiation of new service requirements. In these support activities neural networks can play their role. A few examples are provided below.

- Toll Fraud Detection [30]
- Modeling of Software Metrics [33, 34]
- Preprocessing for Fault Diagnosis Expert Systems [35]
- Character Recognition [36]

1.9 Conclusions

It is clear that neural networks have a large part to play in telecommunications. The parallel computation capability of the networks is attractive in an environment that is of necessity distributed and requires real-time responses. The application of neural networks to telecommunications applications is proceeding by leaps and bounds.

It is clear from the survey presented in this chapter that a lot of the research work in application of neural networks to telecommunications application areas is anecdotal in nature. For example, a researcher may apply a chosen network architecture to a given problem, tweak some parameters or modify the architecture to obtain good learning performance, and report the success of the network in learning a solution to the problem. It is be expected in the years ahead that more emphasis will be put on comparison of neural networks with existing conventional techniques and the derivation of bounds on the performance of neural networks. Given the overlap of pattern recognition with statistical decision theory, well known statistical theory may provide the basis for deriving these bounds.

It is hoped that the reader has seen some of the benefits to be gained from the use of neural networks. Their main drawback is the computation power needed to properly train them. However recent years have seen the cost of computation fall dramatically. It can be expected that neural networks will become increasingly popular and useful in a wide range of application areas.

Bibliography

[1] Hertz, J., Krogh, A., Palmer, R. G., Introduction to the Theory of Neural Computation, Lecture Notes Vol. I, Addison-Wesley, New York, 1991.

[2] Barron, A. R., Universal Approximation Bounds for Superpositions of a Sigmoidal Function, IEEE Transactions on Information Theory, Vol. 39, No. 3, May 1993.

[3] Cybenko, G., Approximation by superposition of a sigmoidal function, Math. Contr. Signals Syst., Vol. 2, pp. 303–314, 1989.

[4] Hornik, K., Stinchcombe, M., White, H., Multilayer feed-forward networks are universal approximators, Neural Networks, Vol. 2, pp. 359–366, 1989.

[5] Littman, M., Boyan, J., A Distributed Reinforcement Learning Scheme for Network Routing, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 45–51, Princeton, New Jersey, November, 1993.

[6] Jensen, J, E, Eshera, M. A., Barash, S., C., Neural Network Controller for Adaptive Routing in Survivable Communications Networks, IJCNN, Vol. II, pp. 29–36, 1990.

[7] Hiramatsu, A., ATM Communications Network Control by Neural Network, IJCNN, Vol. I, pp. 259–266, 1989.

[8] Takahashi, T., Hiramatsu, A., Integrated ATM Traffic Control by Distributed Neural Networks, International Switching Symposium, Vol. III, pp. 59–65, Stockholm, 1990.

[9] Rauch, H. E., Winarske, T., Neural Networks for Routing Communications Traffic, IEEE Control Systems Magazine, pp. 26–31, April 1988.

[10] Ali, M. K., Kamoun, F., A Neural Network Shortest Path Algorithm for Optimum Routing in Packet Switched Networks, pp. 120–124, Vol. 1, Globecom, 1991.

[11] Kwasny, S. C., Kalman, B. L., Engebretson, A. M., Wu, W., Real-Time Identification of Language from Raw Speech Waveforms, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 161–167, Princeton, New Jersey, November, 1993.

[12] Levinson, S. E., Rabiner, L. R., Sondhi, M. M., An Introduction to the Applications of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition, Bell Systems Technical Journal, pp. 1035–1105, Vol. 62, No. 4, April, 1983.

[13] Leerink, L. R., Jabri, M. A., Temporal Difference Learning Applied to Continuous Speech Recognition, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 252–258, Princeton, New Jersey, November, 1993.

[14] Waibel, A., Modular Construction of time-delay Neural Networks for Speech Recognition, Neural Computation, Vol. 1(1), pp. 39–46, 1989.

[15] Lippmann, R. P., Review of Neural Networks for Speech Recognition, Neural Computation, Vol. 1(1), pp. 1–38, 1989.

[16] Fishwick, P. A., Almeida, C., Tang, Z., Time Series Forecasting using Neural Networks vs. Box-Jenkins Methodology, Simulation, 57:5, pp. 303–310, November, 1991.

[17] Box, G. E. P., Jenkins, G. M., Time Series Analysis: Forecasting and Control, rev. ed., Holden–Day, San Francisco, 1976.

[18] Goodman, R. M., Ambrose, B. E., Application of Learning Techniques to Network Management, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 34–44, Princeton, New Jersey, November, 1993.

[19] Connor, J. T., Bootstrap Methods in Neural Network Time Series Prediction, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 125–131, Princeton, New Jersey, November, 1993.

[20] Hsu, Y. Y., Yang, C. C., Design of Artificial Neural Networks for Short Term Load Forecasting, IEE Proceedings-C, pp. 414–418, Vol. 138, No. 5, September 1991.

[21] Peng, T. M., Hubele, N. F., Karady, G. G., Advancement in the Application of Neural Networks for Short-Term Load Forecasting, IEEE Transactions on Power Systems, Vol. 7, No. 1, pp. 250–257, 1992.

[22] Brown, T. X., Neural Networks for Adaptive Equalization, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 27–33, Princeton, New Jersey, November, 1993.

[23] Cid-Sueiro, J., Figueiras-Vidal, A. R., Improving Conventional Equalizers with Neural Networks, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 20–26, Princeton, New Jersey, November, 1993.

[24] Gibson, G. J., Siu, S., Cowan, C. F. N., The Application of Nonlinear Structures to the Reconstruction of Binary Signals, IEEE Trans. Signal Processing, pp. 1887–1884, Vol. 39, No. 8, 1991.

[25] Lo, N. W. K., Hafez, H. M., Neural Network Channel Equalization, Proc. IEEE International Conference on Communications, Vol. II, pp. 981–986, 1992.

[26] Kirkland, W. R., Taylor, D. P., On the Application of Feed-Forward Neural Networks to Channel Equalization, Proc. IEEE/INNS International Joint Conference on Neural Networks, Vol. II, pp. 919–924, 1992.

[27] Milito, R. A., Guyon, I., Solla, S. A., Neural Network Implementation of Admission Control, Advances in Neural Information Processing Systems 3, pp. 493–499, 1991.

[28] Yuhas, B. P., Goldstein, M. H., Sejnowski, T. J., Jenkins, R. E., Neural Network Models of Sensory Integration for Vowel Recognition, Proceedings of the IEEE, 78:1658–1668, 1990.

[29] Yuhas, B. P., Humphries, C., Characterizing DS0-rate Traffic using Neural Networks, IEEE Globecom '92, Orlando, Florida, 1992.

[30] Yuhas, B. P, Toll Fraud Detection, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 239–244, Princeton, New Jersey, November, 1993.

[31] Nordström, E., A Hybrid Admission Control Scheme for Broadband ATM Traffic, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 77–84, Princeton, New Jersey, November, 1993.

[32] Neves, J. E., de Almeida, L. B., Leitão, M. J., ATM Call Control by Neural Networks, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 210–217, Princeton, New Jersey, November, 1993.

[33] Karunanithi, N., Identifying Fault-Prone Software Modules using Connectionist Networks, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 266–272, Princeton, New Jersey, November, 1993.

[34] Carter, S., Frank, R. J., Tansley, D. S. W., Clone Detection in Telecommunications Software Systems: A Neural Network Approach, Proceedings of the International Workshop on Applications of Neural Networks

to Telecommunications, pp. 273–280, Princeton, New Jersey, November, 1993.

[35] Chattell, A., Brook, J. B., A Neural Network Pre-Processor for a Fault Diagnosis Expert System, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 297–305, Princeton, New Jersey, November, 1993.

[36] Amin, A., Al-Sadoun, H., Arabic Character Recognition System using Artificial Neural Network, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, pp. 297–305, Princeton, New Jersey, November, 1993.