Multiple System Artificial Neural Network Model

Kumar Sourav
Software Engineer
Computer Science Corporation
DLF It Park, a-44/45, Sector 62 Noida UP India
sourav.revo@gmail.com

Abstract – This paper is about a connecting neuron named Sron. Sron is a neuron which connects two or more neural network systems and can use the knowledge of one system to further make decisions and produce an outcome. By using this ANN model we can make neural networks for different domains like Finance and Gaming interact and produce some awesome automated systems.

Index Terms— Neural Network, Sron, Multiple System, Model, Automated, Research Robot, Matrix Manipulation.

1. INTRODUCTION

ANN is a human imitated network of neurons which interact with each other to process and transfer information. Out brain consists of a number of neurons which exchange signals with the help of synapses and process data which we encounter every second through vision, touch and other senses and then make a decision based on the input and a previous data of similar experiences. In a similar way we can program neural networks in computer using various programming languages. In biological model every neuron use some bias system which mark information related to as important or less significant.

Similarly we have weights in our simulated computer models which give similar weight to the input we provided to it. So, we can say in all a neural network model has an input state, a number of hidden states and an output state which work to make an intelligent decision. In this paper i would like to introduce modified version of this primitive neural network system which can have applications like : CARR, which is capable of doing research and deriving significant information from various inputs that are given to it. The model consists of an additional neuron called Sron which receives information from output of one or more models and acts as an input to another model. In this way, we are able to incorporate a number of ANN models which can diversify the applications of neural networks. This is like two humans exchange information through some source and makes decisions collectively. Here the source of communication is called Sron. The basic system model is discussed in section 2, Model function is discussed in section 3 and Matrix manipulation is discussed in section 4.

2. BASIC SYSTEM MODEL

Basic System model consists of the traditional ANN model which has an additional Sron neuron connected to the output of a network or a number of networks. The basic ideas is the signal transfer takes place between the output neuron and the Sron in the model. There can be more than one Sron's also but Sron can only exchange information with other neurons and cannot exchange information with Sron's.

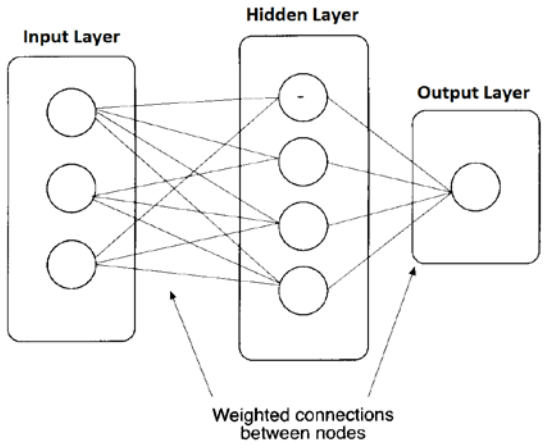

Fig 1 : Basic ANN Model

Figure 1 shows the basic ANN model which has an input layer with any number of neurons, a number of hidden layers can be there with any number of neurons supported by the system for a particular operation and then there is an output layer which takes the weighted sum of all the hidden layer neurons and give an output. Now in Multiple System model the output neuron is connected to the Sron through a Synapse and we can make multiple models to interact with each other.

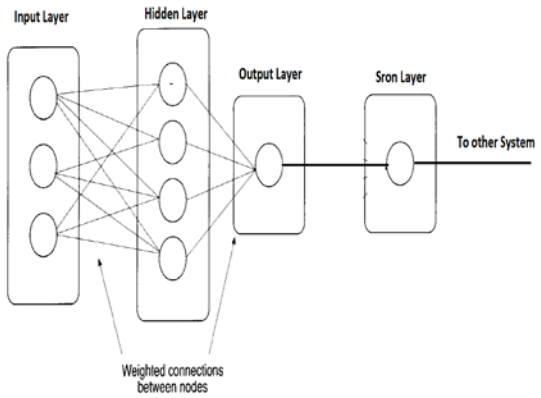

Fig 2 : Multiple System ANN internal Model

Figure 2. Shows how in Multiple Model system the output layer is connected to the Sron layer through a connection. Now we can exploit this connectivity to connect multiple systems together to form a better intelligent system. Suppose we have two scientists develop separate neural networks and we are able to make them communicate them and use them using the Sron layer. We can make more intelligent systems using this technique. The basic overall architecture of Multiple System model is given below.

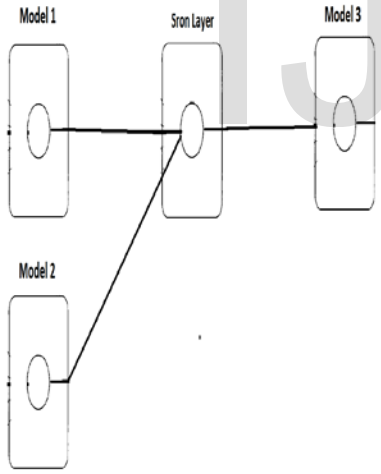

Fig 3 : Multiple System ANN Overall Model

Figure 3. show the overall model of the system. As we can see here System 1 and System 2 which can be a financial neural network or a self trainer or any other model is connected to the System 3 through Sron Layer. Now System 3 takes input from the Sron layer and can start training the system based on prior knowledge gained from the System 1 and System 2. There can be any number of System's and Sron layers. By using this approach we can make a System which can act upon experience of multiple systems, which can lead to a machine which can make decisions for itself.

3. Multiple System Model Functioning

Now we will derive the basic equation and describe the model in form of matrices and see how the system works.
As we can see in Figure 4. an ANN with neurons. Input is given to the neurons which have certain weight associated with them which they use to weight the input given to them. After each input has been weighted , the collective sum goes to the transfer function where we get a summation of the multiplication of all the weights and the input values.

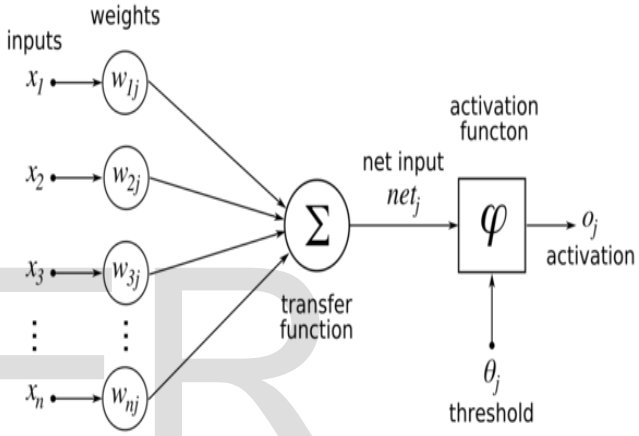

Fig 4 : Neural network with weights and transfer function

Now the equation of the transfer function looks like :

$$Z = \sum_{i=1}^{n} w_i x_i = \text{net}_j \qquad (1)$$

Now, this input is applied to the activation function and we get the following output :

$$O_j = A(Z) \qquad (2)$$

$$O_j = A\left(\sum_{i=1}^{n} w_i x_i\right) \qquad (3)$$

Now the equation (3) is the output generated by one system to the Sron which connects the system to another system.

Every Oj has a switch attached to it which will determine

that will the output be fed to the Sron or not.
Let us denote the switch be Os , depending upon its value is '0' or '1' , the Sron will be fed with the given output of a system.

$$\text{If } O_s = 1 \text{ , } S = \sum_{j=1}^{m} O_j \, w_j \qquad (4)$$

$$\text{If } O_s = 0 \text{ , } S = \sum_{j=1}^{m} O_j \, w_j - K \qquad (5)$$

Where K is the sum of the systems whose output bit value Os is '0'. Here w_j is the weight Sron gives to the output of every ANN system. Now from the given result we can derive the input of another neural network which will act upon the information gathered from the Sron connected systems. Remember, there can be multiple Sron's which will act as input to different neurons of an ANN system.

$$x_k = \sum_{k=1}^{p} S_k \qquad (6)$$

In the equation (6), we have the inputs x_k of the new system which will get the info from the Sron's connected to it Sk. . Each Sron will give input to an individual neuron of the new system , thus forming a more cohesive and intelligent system which will act on the info from the previous attached systems.

Each Sron also has its threshold (Ts) whose value must be exceeded in order for an Sron to be active.

4. Matrix manipulation with Multiple System Model

Now we will derive results for the matrix manipulation for the model. With the help of matrix manipulation we can easily manipulate and make our system train. We can exploit the fact that whole system can be represented in the form of matrices.

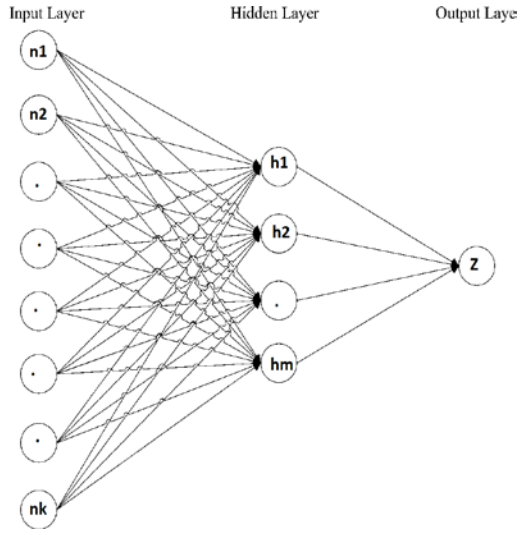

Fig 5 : Neural network model for matrix representation.

In figure 5. we can se the give neural network.
Now we can derive the matrix for Sron as follows.

Let the matrix which connects the input layer neurons and the hidden layer neuron be called NH.

$$NH = \begin{pmatrix} b11 & b12 & .. & b1h \\ b21 & .. & .. & .. \\ .. & .. & .. & .. \\ bn1 & .. & .. & bnh \end{pmatrix} \qquad (7)$$

Now the matrix connecting the hidden neurons with the output neurons be called HO.

$$HO = \begin{pmatrix} O11 \\ O12 \\ .. \\ O1h \end{pmatrix} \qquad (8)$$

Now output will be the multiplication of HO with the weight matrix W.

$$W = \begin{pmatrix} w1 & w2 & .. & wh \end{pmatrix} \qquad (9)$$

Now , the output of each neural network is given by summation of multiplication of weight matrix with HO matrix.

$$O = \begin{pmatrix} O11 \\ O12 \\ .. \\ O1h \end{pmatrix} * \begin{pmatrix} w1 & w2 & .. & wh \end{pmatrix} \qquad (10)$$

We have got the output O of every neural network now, Our next step is to get the input of the Sron which depends on output of every neural network that is connected to it, Sron Weight matrix and the switch of every output.
Let the Switch matrix be X and Sron matrix be S. Remember there is only one switch and Sron matrix associated with each Sron.

$$X = \begin{pmatrix} x1 \\ x2 \\ .. \\ xm \end{pmatrix} \qquad (11)$$

Where m is the number of output switches attached to the outputs of each network.

$$W_s = \begin{pmatrix} w1 & w2 & .. & wm \end{pmatrix} \qquad (12)$$

Here Ws is the Sron weight matrix.

$$S = O^* W_s * X \qquad (13)$$

5. Applications

The model can be applied to a number of fields standalone and to a combinations of different fields.
Some of the applications which this model can offer are :

1) Automated research computer which can do research and make decision based on inputs from multiple models.

2) An automated human assistant who gets inputs daily by the user like : mood , humidity, temperature, heart rate , diet etc and can make best decisions for the user. It can be used with new Google Glass and other personal devices.

3) This model can benefit Astronomy, medicine and other fields with its flexible approach.

6. Conclusion

Multiple System ANN Model is a model which comes with Sron concept. It can be easily implemented in programming and can be accommodated into existing programs by code modifications. This model is a framework for multiple systems to interact and learn.

7. Future Work

This model can be further refined by more detailed analysis and efficiency tests. In near future i would like to do the following work:

1) Implement the model with programming on multiple systems.
2) Incorporate the given model into an existing model and make it link with other models.
3) Do performance tests and analysis.

8. Abbreviations Used

CARR: Computer Automated Research Robot
ANN: Artificial Neural Network

References

[1] Stephen Marsland, Jonathan Shapiro, and Ulrich Nehmzow. A self-organising network that grows when required. *Neural Networks*, 15(8-9):1041-1058, 2002. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2.Oxford: Clarendon, 1892, pp.68–73.

[2] J. L. Shapiro and John Wearden. Reinforcement learning and time perception – a model of animal experiments. *Advances in Neural Information Processing 14*, 2002.K. Elissa, "Title of paper if known," unpublished.

[3] M. Rattray and J. Shapiro. The dynamics of genetic algorithms for a simple learning problem. *Journal of Physics: A*, 29:7451 - 7473, 1996.

[4] *Adam Prügel-Bennett and Jonathan Shapiro. The partitioning problem in unsupervised learning for non-linear neurons.* Journal of Physics A, *26:7417 - 7426, 1993.*

[5] Neil Burgess, M. A. Moore, and J. L. Shapiro. Human-like forgetting in neural network models of memory. In W. K. Theumann and R. Koberle, editors, *Neural Networks and Spin Glasses*. World Scientific,, 1990.