Adaptive Random Fuzzy Cognitive Maps

Jose Aguilar

CEMISID, Dpto. de Computación, Facultad de Ingeniería, Av. Tulio Febres. Universidad de los Andes, Mérida 5101, Venezuela
aguilar@ing.ula.ve

Abstract. The purpose of this paper is to describe an adaptive fuzzy cognitive map based on the random neural network model. The adaptive fuzzy cognitive map changes its fuzzy causal web as causal patterns change and as experts update their causal knowledge. We show how the adaptive random fuzzy cognitive map can reveal implications of models composed of dynamic processes.

1 Introduction

Modeling a dynamic system can be hard in a computational sense. Many quantitative techniques exist. What we seek is a simple method that domain experts can use without assistance in a “first guess” approach to a problem. A qualitative approach is sufficient for this. Fuzzy Cognitive Maps (FCMs) are the qualitative approach we shall take. FCMs are hybrid methods that lie in some sense between fuzzy systems and neural networks[7, 9]. FCMs have gained considerable research interest and have been applied to many areas [4, 9, 10]. However, certain problems restrict its applications. A FCM does not provide a robust and dynamic inference mechanism, a FCM lacks the temporal concept that is crucial in many applications and a FCM lacks the traditional statistical parameter estimates. The Random Neural Network (RNN) has been proposed by Gelenbe in 1989 [5, 6]. This model does not use a dynamic equation, but uses a scheme of interaction among neurons. The RNN has been used to solve optimization and pattern recognition problems [1, 2, 3]. Recently, we have proposed a FCM based on the RNN. The problem addressed in this paper concerns the proposition of an adaptive FCM using the RNN. We describe the Adaptive Random Fuzzy Cognitive Map (ARFCM) and illustrate its application in the modeling of dynamic processes. Our adaptive FCM changes its fuzzy causal web as causal patterns change and as experts update their causal knowledge. In our model, each concept (neuron) is defined by a probability of activation, the dynamic causal relationships between the concepts (arcs) are defined by positive or negative interrelation probabilities, and the procedure of how the cause takes effect is modeled by a dynamic system.

2. Theoretical Aspects

2.1. The Random Neural Network Model

The RNN model has been introduced by Gelenbe [5, 6] in 1989. The model consists of a network of *n* neurons in which positive and negative signals circulate. Each neuron accumulates signals as they arrive, and can fire if its total signal count at a given instant of time is positive. Firing then occurs at random according to an exponential distribution of constant rate r(i). Each neuron *i* of the network is represented at any time *t* by its input signal potential $k_i(t)$. A negative signal reduces by 1 the potential of the neuron to which it arrives (inhibition) or has no effect on the signal potential if it is already zero; while an arriving positive signal adds 1 to the neuron potential. Signals can either arrive to a neuron from the outside of the network or from other neurons. A signal which leaves neuron *i* heads for neuron *j* with probability $p^+(i,j)$ as a positive signal (excitation), or as negative signal with probability $p^-(i,j)$ (inhibition), or it departs from the network with probability d(i). Positive signals arrive to the i^{th} neuron according to a Poisson process of rate $\Lambda(i)$. Negative signals arrive to the i^{th} neuron according to a Poisson process of rate $\lambda(i)$. The main property of this model is the excitation probability of a neuron *i*, q(i), which satisfy the non-linear equation:

$$q(i) = \lambda^+(i)/(r(i)+\lambda^-(i)) \quad (1)$$

where, $\lambda^+(i) = \sum^n_{j=1} q(j)r(j)p^+(j,i)+\Lambda(i)$, $\lambda^-(i) = \sum^n_{j=1} q(j)r(j)p^-(j,i)+\lambda(i)$. The synaptic weights for positive ($w^+(i,j)$) and negative ($w^-(i,j)$) signals are: $w^+(i,j) = r(i)p^+(i,j)$ and $w^-(i,j) = r(i)p^-(i,j)$. Finally, $r(i) = \sum^n_{j=1} [w^+(i,j) + w^-(i,j)]$.

2.2. Fuzzy Cognitive Maps

FCMs combine the robust properties of fuzzy logic and neural networks [7, 9]. A FCM is a fuzzy signed oriented graph with feedback that model the worlds as a collection of concepts and causal relations between concepts. Variable concepts are represented by nodes. The graph's edges are the casual influences between the concepts. In general, a FCM functions like associative neural networks. A FCM describes a system in a one-layer network which is used in unsupervised mode, whose neurons are assigned concept meanings and the interconnection weights represent relationships between these concepts. The fuzzy indicates that FCMs are often comprised of concepts that can be represented as fuzzy sets and the causal relations between the concepts can be fuzzy implications, conditional probabilities, etc. A directed edge E_{ij} from concept C_i to concept C_j measures how much C_i causes C_j. In general, the edges E_{ij} can take values in the fuzzy causal interval [-1, 1] allowing degrees of causality to be represented: i) $E_{jk}>0$ indicates direct (positive) causality between concepts C_j and C_k,. ii) $E_{jk}<0$ indicates inverse (negative) causality between concepts C_j and C_k, iii)

E_{jk}=0 indicates no relationship between C_j and C_k. In FCM nomenclature, model implications are revealed by clamping variables and using an iterative vector-matrix multiplication procedure to assess the effects of these perturbations on the state of a model. A model implication converges to a global stability. During the inference process, the sequence of patterns reveals the inference model.

3. The Dynamic Random Fuzzy Cognitive Maps

Our previous RFCM improves the conventional FCM by quantifying the probability of activation of the concepts and introducing a nonlinear dynamic function to the inference process [3]. The new aspect introduce by the ARFCM is the dynamic causal relationships. That is, the values of the arcs are modified during the runtime of the FCM to adapt them to the new environment conditions. The quantitative concepts allow us develop a feedback mechanism that is included in the causal model to update the arcs. In this way, with the ARFCM we can consider on-line adaptive procedures of the model like real situations. Our ARFCM change their fuzzy causal web during the runtime using neural learning laws. In this way, our model can learn new patterns and reinforce old ones. To calculate the state of a neuron on the ARFCM (the probability of activation of a given concept C_j), the following expression is used [3]:

$$q(j) = \min\left\{\lambda^{+}(j), \max\left\{r(j), \lambda^{-}(j)\right\}\right\} \qquad (3)$$

where

$$\lambda^{+}(j) = \max_{i=1,n}\left\{\min\left\{q(i), W^{+}(i,j)\right\}\right\}$$

$$\lambda^{-}(j) = \max_{i=1,n}\left\{\min\left\{q(i), W^{-}(i,j)\right\}\right\}$$

Such as, $\Lambda(j)=\lambda(i)=0$. That means, the external signal inputs are equal to 0. In addition, the fire rate is $r(j) = \max_{i=1,n}\left\{W^{+}(i,j), W^{-}(i,j)\right\}$. The general procedure of the ARFCM is the following:

1. Define the number of neurons (the number of neurons is equal to the number of concepts).
2. Call the Initialization phase
3. Call the Execution phase.

3.1 The Initialization Procedure

In this phase we must define the initial weights. The weights are defined and/or update according to the next procedures: i) *Based on expert's opinion:* each expert defines its FCM and we determine a global FCM. We use two formulas to calculate the global causal opinion: $E^{G}_{ji} = \max_{e}\left\{E^{e}_{ji}\right\}$, $\forall$ e=1, NE (number of experts); or

$E_{ji}^{G} = \sum_{e=1}^{NE} b_e E_{ji}^{e} / NE$, where E_{ji}^{e} is the opinion of the expert *e* about the causal relationship among C_j and C_i, and b_e is the expert's opinion credibility weight. Then, a) If i≠j and if $E^G_{ij}>0$, $W_{ij}^{+} = E_{ij}^{G}$ and $W_{ij}^{-} = 0$, b) If i≠j and if $E^G_{ij}<0$, $W_{ij}^{-} = E_{ij}^{G}$ and $W_{ij}^{+} = 0$, c) If i=j or if $E^G_{ij}=0$, $W_{ij}^{+} = W_{ij}^{-} = 0$. The causal relationship (E_{ji}^{e}) is caught from each expert from the interval [0, 1]. ii) *Based on measured data:* In this case we have a set of measures about the system. This information is the input pattern: $M=\{D_1, \ldots, D_m\} = \{[d_1^1, d_1^2, \ldots, d_1^n], \ldots, [d_m^1, d_m^2, \ldots, d_m^n]\}$, where d_j^t is the value of the concept C_j measured at time *t*. In this case, our learning algorithm follows the next mechanism:

$$W_{ji}^{t} = W_{ji}^{t-1} + \eta\left(\Delta d_j^t \Delta d_i^t \Big/ \Delta^+ d_i^t \Delta^+ d_j^t \right)$$

where $\Delta d_j^t = d_j^t - d_j^{t-1}$ $\quad$ $\Delta d_i^t = d_i^t - d_i^{t-1}$

$\Delta^+ d_j^t = d_j^t + d_j^{t-1}$ $\quad$ $\Delta^+ d_i^t = d_i^t + d_i^{t-1}$

η is the learning rate.

3.2 The Execution phase

This phase consists on the iteration of the system until the system convergence. The input is an initial state $S_0= \{s_1, \ldots, s_n\}$, such as $q^0(1)=s_1, \ldots, q^0(n)=s_1$ and $s_i \in [0, 1]$ (set of initial values of the concepts). The output $Q^m=\{q^m(1), \ldots, q^m(n)\}$is the prediction of the ARFCM such as *m* is the number of the iteration when the system converge. During this phase, the ARFCM is trained with the next reinforced learning law:

$$W_{ij}^{t} = W_{ij}^{t-1} + \eta\left(\Delta q_i^t \Delta q_j^t\right) \tag{4}$$

where Δq_i^t is the change in the i^{th} concept's activation value among iterations *t* and *t-1*. The algorithm of this phase is:

1. Read input state Q^0
2. Until system convergence
    - 2.1 Calculate q(i) according to the equation (3)
    - 2.2 Update W^t according to the equation (4)

4. Experiment

Dickerson and Kosko proposed a novel use for FCMs [4, 8]. They employed a system of three interacting FCMs to create a virtual reality environment populated by dolphins, fish, and sharks. [9] refines the Dickerson and Kosko's approach to be used the FCM to model the "soft" elements of an environment in concert with an expert system capturing the procedural or doctrinal – "hard" elements of the environment. In their paper, they present a FCM modeling a squad of soldiers in combat. This is a good example where we can use a dynamic model to caught ideas like: an army needs several battles to know the strength of its enemy before a decisive battle. We introduce these aspects in this experiment. The concepts in this map are: i) Cluster (C_1): the tendency of individual soldiers to close with their peers for support, ii) Proximity of enemy (C_2), iii) Receive fire (C_3), iv) Presence of authority (C_4): command and control inputs from the squad leader, v) Fire weapons (C_5), vi) Peer visibility (C_6): the ability of any given soldier to observe his peers, vii) Spread out (C_7): dispersion of the squad, viii) Take cover (C_8): the squad seeking shelter from hostile fire, ix) Advance (C_9): the squad proceeding in the planned direction of travel with the intent of engaging any encountered enemy forces, x) Fatigue (C_{10}): physical weakness of the squad members. In the hybrid system we suggest, the presence of authority concept would be replaced by an input from an expert system programmed with the enemy's small unit infantry doctrine and prevailing conditions. Similarly, the proximity of the enemy would be an input based on the battlefield map and programmed enemy locations. Here, we give them initial inputs and allow them to vary during the runtime of the FCM. The table 1 presents the results for the initial states 0 0 0 1 0 1 1 0 1 0.

Table 1. The edge connection initial matrix for the virtual word experiment.

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}
C_1	0	0	0	0	0	1	-1	0	0	0
C_2	1	0	1	0	1	0	0	1	0	0
C_3	1	0	0	1	-0.1	0	0	1	0	1
C_4	-1	0	0	0	0	0	1	-1	1	0
C_5	0	-0.5	-0.12	0	0	0	0	0	0	0.2
C_6	0	0	0	0	0	0	0	-0.7	1	0
C_7	-1	0	-0.5	0	0	0	0	0	0	0
C_8	1	0	0	1	-0.7	1	0	0	-1	-1
C_9	0	1	0	0	0	0	0	0	0	1
C_{20}	0	0	0	0	-0.5	0	0	0	0	0

Table 2. The results for the virtual word experiment.

Input	Kosko FCM	DFRCM	Iteration
0 0 0 1 0 1 1 0 1 0	0 0 0 1 0 1 1 0 1 0	0.2 0.4 0.7 0.6 0.5 0.6 0.6 0.4 0.6 0.4	1
	1 1 1 1 0 1 0 1 0 1	0.6 0.6 0.6 0.6 0.5 0.1 0.4 0.6 0.6 0.8	2 *
	1 0 1 1 0 1 0 1 1 0	0.6 0.6 0.6 0.6 0.5 0.1 0.4 0.6 0.8 0.8	3
	1 1 0 1 0 1 0 0 1 1	0.8 0.8 0.6 0.6 0.8 0.1 0.2 0.8 1 0.8	4
	0 1 1 0 1 1 0 0 1 1	1 1 0.8 1 0.8 0 0 0.8 1 0.8	5
	0 1 1 1 0 0 0 0 1 1		6
	1 1 1 1 1 0 0 1 1 1		7

We define the starting state S_0=(0 0 0 1 0 1 1 0 1 0) i.e., presence of authority, peer visibility, spread out and advance are present, but all other concepts are inactive. The system stabilizes to the state S_7 (Kozko model) or state S_5 (ARFCM). The introduction of new information during the runtime doesn't affect the convergence of our system (we obtain the same result of Kozko). This is reasonable system operation and suggests the feasibility of FCMs as simple mechanisms for modeling inexact and dynamic behavior that is difficult to capture with formal methods.

4. Conclusions

In this paper, we have proposed an adaptive FCM based on the RNN, the ARFCM. We show fusing the RFCM with a traditional reinforced learning algorithm can yield excellent results. The ARFCM may be rapidly adapted to changes in the modeled behavior. It is a useful method in complex dynamic system modeling. We do not observe any inconsistent behavior of our ARFCM with respect to the previous FCMs. Our ARFCM exhibit a number of desirable properties that make it attractive: i) Provide qualitative information about the inferences in complex dynamic models, ii) Can represent an unlimited number of reciprocal relationships, iii) Is based on a reinforced learning procedure, iv) Facility the modeling of dynamic, time evolving phenomena and process, v) Has a high adaptability to any inference with feedback. Another important characteristic is its simplicity, the result of each ARFCM’s cycles is computed from the equation (3). The ease of construction and low computational costs of the ARFCM permits wide dissemination of low-cost training aids. In addition, the ability to easily model uncertain systems at low cost and with adaptive behavior would be of extraordinary value in a variety of domains.

References

1. Aguilar, J. Definition of an Energy Function for the Random Neural to solve Optimization Problems, Neural Networks, Vol. 11, Pergamo, (1998) 731-738.
2. Aguilar, J. Learning Algorithm and Retrieval Process for the Multiple Classes Random Neural Network Model. Neural Processing Letters, Vol. 13, Kluwer Academic Publishers, (2001) 81-91.
3. Aguilar, J. A Fuzzy Cognitive Map Based on the Random Neural Model. Lecture Notes in Artificial Intelligence, Vol. 2070, Springer-Verlag, (2001) 333-338.
4. Dickerson J., Kosko B. Virtual Worlds as Fuzzy Dynamic Systems. in Technology for Multimedia, (B. Sheu, editor), IEEE Press (1996).
5. Gelenbe E. Random neural networks with positive and negative signals and product form solution, Neural Computation, Vol. 1, (1989) 502-511.
6. Gelenbe E. Theory of the random neural network model, Neural networks: Advances and Applications (E. Gelenbe editor), North-Holland, Pays-Bas (1991).
7. Kosko B. Fuzzy Cognitive Maps, Int. Journal of Man-Machine Studies, Vol. 24, (1986) 65-75.
8. Kosko B., Dickerson J. Fuzzy virtual worlds. AI Expert, (1994) 25-31.
9. Kosko B. Fuzzy Engineering, Prentice-Hall, New Jersey (1999).
10. Miao Y., Liu C. On causal inference in Fuzzy Cognitive Map, IEEE Transaction on Fuzzy Systems, Vol. 8, (2000) 107-120.