PalArch's Journal of Archaeology of Egypt / Egyptology

Pattern Classification Using ART 1 Network

Chanda Thapliyal Nautiyal[1], *Sunita Singh*[2], *U.S. Rana*[3]

[1] Government Degree College, Narendra Nagar, Uttarakhnd, India
chanda.nautiyal@gmail.com

[2] IRDE(DEAL), Dehradun, India
sunnidma@yahoo.com

[3] DAV(PG) College, Dehradun, India
drusrana@yahoo.co.in

Chanda Thapliyal Nautiyal[1], Sunita Singh[2], U.S. Rana[3], Pattern Classification Using ART 1 Network - Palarch's Journal Of Archaeology Of Egypt/Egyptology 17(9). ISSN 1567-214x, Keywords: Artificial neural network, ART1 Neural Network, back propagation, pattern classification, stability-plasticity dilemma

Abstract-
Conventional artificial neural networks are unsuccessful in solving the plasticity- stability dilemma. For a situation where there is real network, this is exposed to regularly varying environment; it is possible that it might not get the similar training vector twice ever again [22]. Going through these circumstances, back propagation will learn nothing. In this paper pattern classification is done using ART1 network. Three different sets of seven alphabets are taken as input. The proposed algorithm successfully classifies 57.14% characters. The number of maximum clusters formed is 15 and vigilance parameter is 0.5.

1 Introduction

All those intelligent systems which can autonomously acclimatize in real time to the unpredicted change in the world confront the basic design problem. The problem is stability- plasticity dilemma. Adaptive resonance theory ART gives answer to this problem [19, 20].

ART or Adaptive Resonance Theory describes class of self-organizing neural architectures which clusters the pattern space and then constructs suitable weight vector templates [10, 12]. A typical artificial neural network is not competent in solving the stability-plasticity dilemma. A network should be open for new learning (should remain plastic) without removing formerly learned codes [9, 11, 12]. If training vectors are only fixed sets, the network might be driven up throughout them over and over again and can finally learn all [22]. For a situation where there is real network, this is exposed to regularly varying environment; it is possible that it might not get the similar training vector twice ever again [22]. In these situations, back propagation cannot learn anything. ART algorithms and networks retain the plasticity. Patterns are not returned to a previous cluster by a stable net. Stability is achieved by some nets by slowly minimizing the learning rate, because the identical bunch of instruction structure is being presented many times [23]. This capability of a net is termed as plasticity i.e. (learning a new pattern smoothly at whichever point of training). Modeling of ART nets is plastic besides being stable.

The ART network is vector categorizer [24]. After taking the input vector ART network categorizes it in the group where it matches the stored patterns the most [24]. If the input vector is different from the saved pattern, a new group is created by storing pattern that is like the input vector [8, 9, 22, 24]. Once the stored pattern is found matching the input vector inside a prescribed tolerance, such pattern is made to modify (same as that of input vector) [8, 9, 22, 24]. Otherwise not. In ART, differential equations govern the changes in the activation of units and weights [8, 9, 10]. When an admissible cluster unit is chosen for learning, the weights may be retained over a prolonged duration [11, 12]. During this period weight changes should be done only. This period is called 'resonance period'. Thus the net gets the name 'resonance' in it [11, 12].

ART networks and algorithms retain the flexibility necessary for grasping fresh patterns, just when formerly adapted patterns are not changed [9, 13].

2 Literature Survey

Jain et al. (2014) proposed a solution to the problem of forgery detection and automatic signature verification [2]. A Handwritten Hindi character recognition technique was proposed by Tanuja et al. (2015) [7] by Canny Edge Detection technique as well as artificial neural network. Another approach for character recognition includes firefly based back-propagation network (Sahoo et al. (2015) [5]. Handwriting digits' recognition for South Indian languages was worked upon by Pauly et al. (2015) applying ANN [4]. A Malaysian's car plate's recognition system was described using traditional backpropagation algorithm (BPP) (On et al. (2016)) [3]. Sethy and Patra (2016) attempted building up handwritten Odia number digits' recognizer [6]. With 400 input neurons, recognition was 80.2% and 90% for binarization & DCT respectively [6]. Afroge et al. (2016) presented an ANN based method for the recognition of English characters [1]. The network had 96 input neurons and 280 epochs.

In this paper pattern classification is done using ART1 network. The proposed algorithm successfully classifies 57.14% characters. The number of maximum clusters formed is 15 and vigilance parameter is 0.5. Paper is organized as follows: Next section describes ART1 architecture and algorithm. Then the paper specifies simulation results and the paper concludes with conclusion.

3 ART 1

Architecture

ART 1 network embodies computational units &supplemental units. Architecture is shown in fig. 1 [14].

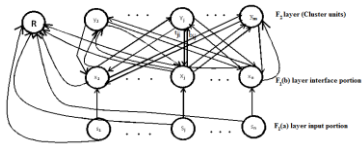


Fig. 1. Architecture of ART 1

3.1 Computational Unit

The computational unit has F1, F2 units & reset mechanism unit also. The input portion (F1(a)) is inked with interface portion (F1(b)) [13]. (F1(a)), (F1(b)) are linked with reset mechanism unit with the help of weighted bottom-up connections and weighted top-down connections [13,24].

3.2 Supplemental Unit

Difficulty posed in computational units (F1, F2 and reset unit): those units expected to react separately for separate phases of task [9, 13]. Also course of action of reset mechanism put up a difficulty that, F2 units ought to be restricted/prevented for some conditions, but should also come back for obtainability on succeeding learning [10, 13, 24].

Each of these difficulties may be resolved upon installation of two supplemental units (gain control units, G1 and G2), besides ‘R’ (Figure 2). Every unit out of these 3 exclusive units will receive

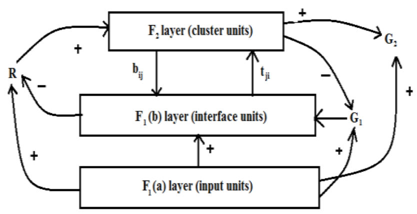


Fig. 2. Structure of Supplemental Units

signals coming from each and every unit of the layers moreover also delivers their signals to each and every unit of layers (Figure 2). “+” represents Excitatory signal and “—" represents inhibitory signal [15, 23]. When, in the allocated layer some unit is "on", signal might be sent. All the units of the interface region (F 1 (b)) and F2 layer in ART1 net can gather a

signal from the three sources. Meanwhile options for interface units(F 1 (b)) for obtaining signals are an input signal (F 1 (a) unit), a top-down signal (F2, node), and the G1 unit. In a similar way, options for F2 unit for obtaining signals are from unit R, from interface units F1 or from the G2 unit. To be "on" either F2 unit or an interface unit F1 (b) have to obtain two excitatory signals. The availability of three alternative signals' sources makes this called "two-thirds rule". Initial weights and parameters can be chosen with the help of two-thirds rule. As shown in Figure 2, reset unit R is the controller of matching of vigilance [9, 10, 13, 15, 18, 23].

4 Algorithms

Art1 algorithm goes like this [15, 16, 17]. The notations used (in the algorithm) are

m - Maximum clusters what can be formed, n - Components in the input vector, s - binary input vector, x - for interface (F_1(b) layer (binary)), activation vector, ρ - vigilance parameter, b_{ij} - weights (bottom-up from F_1(b) to F_2 unit), t_{ji} - weights (top-down from F_2 to F_1(b) unit), ||x|| - norm of vector x.

Description [15, 16, 17, 18, 22].

F_1(a) layer is offered **s** (binary input vector). Then signals are acquired by F_1(b). The activation signal is sent by F_1(b) layer to F_2 layer with bottom-up weighted interconnected paths. Total input is then calculated by every F_2 unit. The activation is fixed to 1 for the unit that has the maximum net input (winning unit) .The activation for others units will be remained zero. Present input pattern will solitarily be learnt by the winning unit. A signal transmitted downwards (multiplied by top-down weights) from F_2 to F_1(b). When 'X' unit receives non-zero weights from F_1(a) as well as from F_2 units , it remains 'on' [18, 22]. Total constituents for which **'s'** (input vector) as well as **'t_j'** are one, is norm of the vector x, ||x||. Weights adjustments for winning cluster unit are done on the basis of proportion (||x||/||S||) (Match Ratio). The entire procedure is carried on till an acceptable match or inhibitions of the all neurons [18, 22].

Finally after entire weights adjustments, whole cluster units come back to '0' activation (inactive status) meanwhile they are open to participate further [18, 22].

The training algorithm of ART 1 [15, 22] is as follows.

- Step I: Initialize parameters: $0 < \rho \leq 1$. L > 1,Initialize weights: $0 < b_{ij}(0)$ $<L/(L-1+n)$, $t_{ji}(0)=1$
- Step II: perform steps from III to XIV, till the stopping condition becomes false.
- Step III: perform Steps from IV to XIII for every training input.
- Step IV: F_2 units' activations are fixed as '0'. And F_1(a) units' **s (**input vector).
- Step V: norm **s** is computed as: $||s||=\sum_i S_i$
- StepVI: $x_i= s_i$ input signal Sent ; F_1(a) to F_1(b) layer.

- Step VII: every F_2 node which is not inhibited: $y_j = \sum_i b_{ji} x_i$ if $y_J \neq -1$
- Step VIII: perform step from IX to XII, till reset condition becomes true.
- Step IX: J will be spotted for nodes j with condition $y_J \geq y_j$. Pattern is not possible to cluster for $y_J = -1$ and then all nodes are inhibited. Step X: $x_i = s_i t_{Ji}$ For F_1(b) x (activation) is computed again :
- Step XI: vector **x** norm is Computed as : $||x|| = \sum_i X_i$
- Step XII: Reset test: $y_J = -1$ (inhibit node J) (and Step VIII continued to be executed again) if $||x||/||s|| < \rho$,otherwise if $||x||/||s|| \geq \rho$, go to the next Step XIII.
- Step XIII: weights for node J are updated: $b_{ij(new)} = Lx_i / L - 1 + ||x||$ $t_{Ji(new)} = x_i$
- Step XIV: stopping condition is tested.

Stopping condition in Step XIV is one out of three: 1.No reset of units. 2. No change in weight. 3. Maximum number of epochs arrived at.

All former learning trial obstructions are cleared in Step IV (presenting a pattern).

In Step IX, J is to be taken as the least such index, if there is a tie. This is to be noted that t_{ji} can only take two values either 1 or 0, and if during the learning process t_{ji} is fixed at 0, there is no chance of it being set back to 1 (that is why stable learning method is arrived at) [15, 22].

Typical values of the used parameters [15, 18]. ρ=0.9, L=2, b_{ij} =1/(1+n), t_{ji}=1

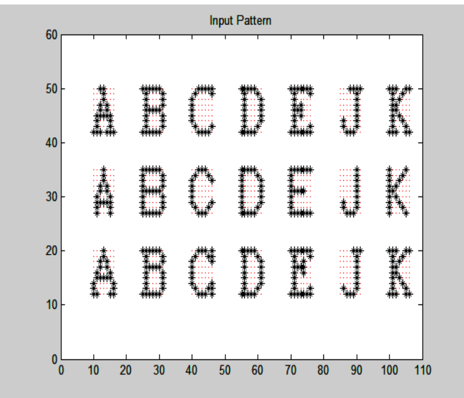
5 Methodology

A data file is formed for the input pattern. Seven alphbets 'A, B, C, D, E, J, K' are taken as input. Three different sets of each alphabet is taken. For the modelling of each alphabet, a 9 x 7 dot array is constructed. Symbol '*' indicates a '+1', if it is present, else it represents '-1'. Each alphabet has a unique pattern. In the same way three different patterns are taken. These patterns are presented as input to the ART 1 network according to the above algorithm. The number of maximum clusters formed is 15 and vigilance parameter is 0.5. The algorithm has been simulated in MATLAB for pattern classification.

6 Simulation Results and Discussion

Simulation results are executed for the classification of English alphabets 'A, B, C, D, E, J, K' in Matlab. Three different sets of alphabets are presented as inputs. After single epoch the algorithm classifies each alphabet as per the following.

A=66.67%, B=0%, C=66.67%, D=100%, E=66.67%, J=66.67%



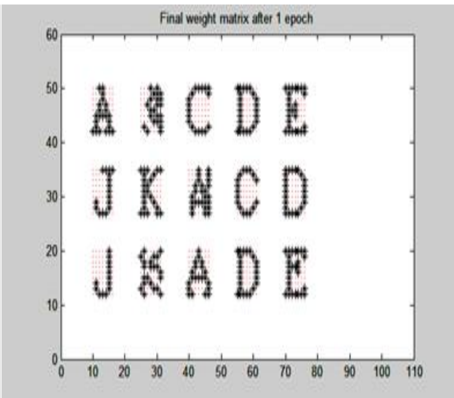


Fig. 3: Input pattern for training characters Fig. 4: Final weight matrix after

single epoch

7 Comparison With State of Art Techniques and Conclusion

Author	**Neurons in input Layer,Number of epochs**	**Recognition Accuracy**
Jain et al. (2014)	13 inputs and 100 epochs	51.7% to 97.9%
Tanuja et al. (2015	1500 epochs	90% to 100%
Sahoo et al. (2015		52.94% to 94.11%.
Pauly et al. (2015	3780 neurons	3780 neurons 83.4%
On et al. (2016)	9121 and 9927	9121 and 9927 epochs 96% to 98%
Proposed Algorithm	63 neurons and 1 epoch	33.33% to 100% noise

The proposed algorithm has been tested successfully for pattern classification of seven letters of English alphabets. ART 1 network with 15 maximum numbers of clusters and 0.5 vigilance parameter classifies alphabets with accuracy ranging from 33.33% to 100% (average accuracy 57.14%). This algorithm may further be extended for different types of objects

References

1. Afroge, S., Ahmed, B., & Mahmud, F. (2016, December). Optical character recognition using back propagation neural network. In Electrical, Computer & Telecommunication Engineering (ICECTE), International Conference on (pp. 1-4). IEEE.
2. Jain, C., Singh, P., & Chugh, A. (2014). An Offline Signature Verification using Adaptive Resonance Theory 1 (ART1). International Journal of Computer Applications, 94(2).
3. Kim On, C., Kein Yau, T., Alfred, R., Teo, J., Anthony, P., & Cheng, W. (2016). Backpropagation Neural Ensemble for Localizing and Recognizing Non-Standardized Malaysia's Car Plates. International Journal on Advanced Science, Engineering and Information Technology, 6(6), 1112-1119.
4. Pauly, L., Raj, R. D., & Paul, B. (2015, August). Hand written digit recognition system for South Indian languages using artificial neural networks. In Contemporary Computing (IC3), 2015 Eighth International Conference on (pp. 122-126). IEEE.
5. Sahoo, M. K., Nayak, J., Mohapatra, S., Nayak, B. K., & Behera, H. S. (2015). Character recognition using firefly based back propagation neural network. In

Computational Intelligence in Data Mining-Volume 2 (pp. 151-164). Springer, New Delhi.
6. Sethy, A., & Patra, P. K. (2016, April). Off-line Odia handwritten numeral recognition using neural network: A comparative analysis. In Computing, Communication and Automation (ICCCA), 2016 International Conference on (pp. 1099-1103). IEEE.
7. Tanuja, K., Kumari, V. U., & Sushma, T. M. (2015). Handwritten Hindi Character Recognition System Using Edge detection & Neural Network. International Journal of Advanced Technology and Engineering Exploration, 2(6), 71.
8. S. Rajakarunakaran, P. Venkumar, D. Devaraj, K. Surya Prakasa Rao. "Artificial neural network approach for fault detection in rotary system",Applied Soft Computing, 2008
9. Braspenning, Petrus J., Frank Thuijsman, and Antonius Jozef Martha Maria Weijters. Artificial neural networks: an introduction to ANN theory and practice. Vol. 931. Springer Science & Business Media, 1995.
10. Yegnanarayana, Bayya. Artificial neural networks. PHI Learning Pvt. Ltd., 2009.
11. https://www.tutorialspoint.com/artificial_neural_network/artificial_neural_network_quick_guide.htm
12. www.ijecce.org
13. www.academypublisher.com
14. baadalsg.inflibnet.ac.in
15. Fausett, L. (1994). Fundamentals of neural networks: Architectures,
16. Gurney, Kevin. An introduction to neural networks. CRC press, 1997.
17. Zurada, Jacek M. Introduction to artificial neural systems. Vol. 8. St. Paul: West, 1992.
18. www.scribd.com
19. Gurney, Kevin. An introduction to neural networks. CRC press, 1997.
20. Carpenter, G.A. and Grossberg, S., 1988. The ART of adaptive pattern recognition by a self-organizing neural network. Computer, 21(3), pp.77-88.
21. Chanda Thapliyal Nautiyal, Sunita Singh, U. S. Rana. "Chapter 13 Recognition of Noisy Numbers Using Neural Network" , Springer Science and Business Media LLC, 2018.
22. S. Rajakarunakaran, P. Venkumar, D. Devaraj, K. Surya Prakasa Rao. "Artificial neural network approach for fault detection in rotary system" , Applied Soft Computing, 2008
23. edoc.pub
24. www.academypublisher.com Anitha. S.: Lossless Image Compression and Decompression Using Huffman Coding. In: International Research Journal of Engineering and Technology (IRJET), 2(1), pp. 240-247 (2015)