Flagged & Compact Fuzzy ART: Fuzzy ART in more efficient forms.

Kamal R. Al-Rawi [1] , and Consuelo Gonzalo [2] ;

Department of Computer Science, Faculty of Information Technology, Petra University,

Amman, Jordan.

DATSI, Facultad de Informática; Universidad Politécnica de Madrid, Spain.

e-mail: kamalr@uop.edu.jo and chelo@fi.upm.es

Abstract: Two new simplified algorithms for Fuzzy ART have been developed. Only committed category nodes C rather then the full capacity of the category nodes N (N>>C) are involved in determining the winning categoy node. In addition to that, the initialization for weights and choice values has been eliminated, and the calculation of $\sum_{i=1}^{2M} A_i$ is replaced by M, since $A_i + A_i^- = 1$.

This reduces a lot the training time without altering the categorazation accuracy. While the new architectures are presented toward the fuzzy ART ANN, in this work, it can be applied to all ART ANNs.

Keywords: Compact fuzzy ART, Flagged fuzzy ART, ART ANN, Unsupervised ART, and Unsupervised learning.

1. Introduction

The Fuzzy ART (Adaptive Resonance Theory) is an unsupervised ART-based ANN. Its architecture has been designed for learning and categorization of arbitrary analog or binary multi-valued input patterns.

Input patterns ($A_i^{(t)}, i = 1,...,M$) $\in [0,\ 1]$ are presented with its complement at the input layer F_1. The choice function $T_j^{(t)}, j = 1,...,C$ for each committed category node of the category layer F_2 is computed:

$$T_j^{(t)} = \frac{\sum_{i=1}^{2M}(A_i \wedge w_{ij})}{\alpha + \sum_{i=1}^{2M} w_{ij}}; j = 1,...,C \quad \text{..1}$$

Where C is the number of committed category nodes, w_{ij} the weight that connect the input node "i" with the committed category node "j", and α>0, the choice value parameter.

The choice value $T_j^{(t)}$ represents the activation level of each committed category node.

The winning committed category node is determined. It represents the category node with the highest choice value among all category nodes N in the category layer:

$$T_J^{(t)} = \max\{T_j^{(t)}; j = 1,...,N\} \quad \text{...2}$$

The value of *N* is normally much larger than *C* (*N*>>*C*). All category nodes *N* are involved, which has been employed by (Carpenter *et al.* 1991), instead of committed category nodes *C*. Their reasoning for this is to let uncommitted category nodes be committed, when it is required, in a sequence order (1, 2, ..., *j*-1, *j*, *j*+1, ..., *N*). To achieve this, they assigned a very small positive value ϕ_j to each category nodes before training is started. They called it*, "F_2-order constants"*. These values are decreasing as the index of the order of category node *j* in the memory field is increased.

$$0 < \phi_N < ... < \phi_j < ... < \phi_1 \cong 0 \quad \text{...3}$$

In this way, when all committed category nodes are in shut off mode, because they failed to represent the current input, the uncommitted category node $(C+1)^{(t)}$ will be committed, since it has the highest choice value (F_2*-order constant*) among all uncommitted category nodes as prearranged. They assigned values near zero assuming that there is no computed choice value, for any committed category node, less than ϕ_1.

The match value is computed for the winning category node J:

$$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{\sum_{i=1}^{2M} A_i} \qquad \text{.........4}$$

The match value represents a hypothesis, that the current input $A^{(t)}$ belong to the winning category node J of the F_2-field. This hypothesis is tested against predetermined vigilance parameter $\rho \in [0,\ 1]$. The vigilance parameter represents the minimum confidence level that is required to accept that the winning node J of the F_2-field, represents the category of the current input $A^{(t)}$.

If the match value of the winning node is less than ρ, the hypothesis is rejected and this committed category node shuts off as far as the current input is presented to the network. This is to prevent the persistent selection of the same category node during search. Shut off is simply done by assigning -1 to the choice value of the failed category node. Researching for another winning committed category node is triggered among all category nodes N. The network keeps searching for maximum choice value node J, doing computation of the match function for node J, and testing against the vigilance parameter ρ, for each committed node of the F_2-field. This is done in order, according to their choice values' rank, until either one of the committed category node can represent the current input $A^{(t)}$ (resonance occurs), then learning the weight w_{iJ} of the selected category node J, or if none, the uncommitted category node with index C+1, which has the highest choice value among all uncommitted category nodes as prearranged, will be picked by the network to represent the current input. The match value of new committed category node passes the value vigilance parameter ρ, since it has the value of one.

The weights of the selected category node are updated in order to incorporate the characteristics of the input pattern to category *J*:

$$w_{iJ}^{new} = \beta(A_i \wedge w_{iJ}^{old}) + (1-\beta)w_{iJ}^{old} \quad \text{...5}$$

After learning the weights of the selected category node *J*, a check should be done to see if a committed category node has been chosen to represent the current input or a new category node has been committed. This is to increase the number of committed category node *C* by one or not. C controls the computation of the choice function for committed category nodes only.

If the index of the selective category node *J* is greater than *C* that means the uncommitted category node *C*+1, has been chosen to represent the current input, because all committed category nodes failed to do so. This uncommitted category node has the maximum prearranged choice value ϕ_{C+1} among the choice values ϕ_j of all other uncommitted category node, since it is prearranged so. The full architecture of the Fuzzy ART is shown in (figure-1).

2. Newly developed versions of Fuzzy ART

The determination of the winning category node among the full capacity of the network *N*, as reported by (Carpenter & Grossberg 1987, Carpenter *et al.* 1991), is time consuming. The capacity of the system can be very large especially when it is working in a non-homogenous environment. Uncommitted category nodes can be committed in sequential order without using F_2*-order constants* (the prearranged choice values ϕ_j) and without including all the capacity of the category layer *N* in determination the maximum choice value node *J*.

Two simplified versions of Fuzzy ART architectures have been presented in this work called "*Flagged Fuzzy ART" ,* and *"Compact Fuzzy ART"*. The first approach involves the uncommitted category node with rank C+1 in the category layer together with all committed category nodes to determine the maximum choice value node J. A total of C comparison is required rather than N-1 as the case in the original Fuzzy ART architecture. While in the second approach, the uncommitted category nodes only are involved in determination of the maximum choice value node. This requires C-1 compressions for a current exemplar. We have to remember that N>>C.

2.1 Flagged Fuzzy ART

2.1.1 The architecture of Flagged Fuzzy ART

There is no reason at all to involve T_{C+2} , ..., T_N in determination the maximum choice value node. Only the uncommitted category node with rank C+1 in the category layer will be involved. This uncommitted category node is flagged by assigning a value of ϕ_{C+1} to its choice value such that;

$$T_{shut-off} < \phi_{C+1} < 0 \quad \text{...6}$$

A negative value is assigned for ϕ_{C+1}, because the input features A_i as well as the weights w_{ij} never have negative values. So, the choice value for any committed category node is never a negative value,

$$T_j \geq 0 \quad ; j=1, ..., C \text{ .... 7}$$

However, the value of ϕ_{C+1} must be greater than the choice value of committed category nodes that are in shut off mode. In this way, when all committed category node are in shut-off mode, the flagged node with index C+1 in the category layer, will be chosen as the maximum choice value node. We should not worry about the match value of a new committed category node, since the match value of any new committed node is equal to one, which is the highest value that the vigilance parameter ρ can have. That is because A_i is normalized to [0, 1] before its presentation to the network, and the initial weights for category nodes are equal to one. So input A_i is a subset of $w_{i,C+1}$. That means $A_i \wedge w_{i,C+1} = A_i$. Computing the match function for the subset choice leads always to one as demonstrated below:

$$\frac{1}{M}\sum_{i=1}^{2M}(A_i \wedge w_{i,C+1}) = \frac{1}{M}\sum_{i=1}^{2M} A_i = \frac{M}{M} = 1 \quad \text{...8}$$

Therefore, the uncommitted flagged node C+1 will not go to shut off mode. It will pass the match test for sure.

The choice value is computed as that for fuzzy ART (equ.1). Determination of the maximum choice value node is among C+1 nodes rather than the full capacity of the category node N. The match value is:

$$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{M} \quad \text{..9}$$

The denominator is "M" here rather than $\sum_{i=1}^{2M} A_i$ as in (equ.4), since $A_i + A_i^- = 1$.

After resonance occurs, a check should be done to see if the flagged uncommitted category node is chosen. If $J>C$ then the flagged node has been chosen. The number of committed category node must be increased by one (C=C+1) and the weights of the new flagged node $w_{i,C+1}$ should be initiated;

$$w_{i,C+1} = 1 \quad ; i=1, ..., 2M \quad \text{..10}$$

The full architecture of the *Flagged*-Fuzzy ART is shown in (figure-2). Only the flagged uncommitted category node in addtion to the committed category nodes are involved in determination the maximum choice value node.

2.1.2 Training algorithm of Flagged Fuzzy ART

1) Input parameters;

a) Dynamic parameters:

i- $\rho \in$ (0, 1]: The vigilance parameter. Note that $\rho \neq 0$

ii- $\beta \in$ (0, 1]: The dynamic learning parameter; $\beta = 1$ for fast learning. note that $\beta \neq 0$

iii- α >0: The choice value parameter. This parameter is used to break the tie for the choice values toward the most probable category node to represent an input patterns. However, it can be eliminated since such occurrence is rare, and re finding the maximum choice value node is required much less works than the original fuzzy ART.

b) Data characteristics;

i- M: The dimension of the input features.

ii- Pt: The number of exemplars to be used in learning.

c) Initialization;

i- Number of iteration t=1.

ii- Number of committed category nodes C=1.

iii- $T_{C+1} = -0.1$

2) New input;

$$A_i^{(t)} = \begin{cases} a_i^{(t)} & for\ \ 1 \le i \le M \\ 1 - a_i^{(t)} & for\ \ M + 1 \le i \le 2M \end{cases}$$

3) Compute the choice function for all committed category nodes;

$$T_j^{(t)} = \frac{\sum_{i=1}^{2M}\left(A_i^{(t)} \wedge w_{ij}\right)}{\alpha + \sum_{i=1}^{2M} w_{ij}} \quad , j=1, ..., C$$

4) Reset: Determine the node *J*, which has the maximum choice value;

$T_J^{(t)} = max\{T_j^{(t)}\}$, *j*=1,..., *C*+1

5) Matching criterion: If ($\sum_{i=1}^{2M}(A_i^{(t)} \wedge w_{iJ})/M < \rho$) then;

i- Shut off this node to put it out of competition;

$T_J^{(t)} = -1$

ii- GOTO STEP (4)

6) If (*J*>*C*) Then new category node has been committed

i- *C*=*J*

ii- $w_{iJ} = 1$; *i*=1, ..., *2M*

iii- $T_{C+1} = -0.1$

7) Training;

$$w_{iJ}^{new} = \beta(A_i^{(t)} \wedge w_{iJ}^{old}) + (1-\beta)w_{iJ}^{old}$$

8) If (t<Pt) Then;

i- t=t+1

ii- GOTO STEP (2)

9) Training has been done. The network is ready for categorization.

2.2 Compact Fuzzy ART

2.2.1 The architecture of Compact Fuzzy ART

Uncommitted category nodes can be committed in sequential order without using even the flagged uncommitted category node. It involves only the committed category node to determine the maximum choice value node *J*.

The choice function is computed for committed category nodes. The maximum choice value node *J* is determined among committed category nodes *C* only.

$$T_J^{(t)} = max\{T_j^{(t)}\}\ ; j=1, ..., C \quad \text{...11}$$

The match value of the selected category node *J* is tested against the predetermined value of the vigilance parameter ρ. If the match value of node *J* is less than ρ, the node is shut off by assigning a value of –1 to its choice value to put it out of competition during the current input. Otherwise, the node is trained, all committed category nodes are on, and new input is presented to the network.

When the maximum choice value equals –1 all committed category nodes are in shut off mode. The uncommitted category node *C*+1 should be committed to represent the current input in order to prevent the fragmentation of the category layer. Simply training the initial weights of the category node with index *C*+1, and increasing the count of the committed category nodes by one can do this. This commits the uncommitted category nodes according to their order in the category layer. The number of comparison needed to determine the maximum choice value node is (*C*-1) rather than (*N*-1) which the original Fuzzy ART algorithm is required. This will save a lot of computation time, keeping in mind that *N*>>*C*.

In the case of new category node should be committed, its weights will be updated through the next equation:

$$w_{i,C+1}^{first} = \beta A_i^{(t)} + (1-\beta) \quad ; i=1, ..., 2M \quad ……………………...12$$

According to this equation weights initialization (w_{ij} ; i=1, ..., 2M; j=1, ..., N) is not required, as reported by (Carpenter *et al*. 1991). It is implemented implicitly in the training equation itself. This will save time since this equation requires less arithmetic operations. The full architecture of *Compact*-Fuzzy ART is shown in (figure 3). Committed category nodes are shown in dark. Uncommitted category nodes are shown in light. Weights connect all input layer nodes to committed category nodes only. Weights are not connected to uncommitted category nodes since they are not committed yet (they are not assigned weights yet).

2.2.2 Training algorithm of Compact Fuzzy ART

1) Input parameters;

a) Dynamic parameters;

i- $\rho \in$ (0, 1]: The vigilance parameter.

ii- $\beta \in$ (0, 1]: The dynamic learning parameter; $\beta = 1$ for fast learning.

iii- α >0: The choice value parameter. It can be eliminated.

b) Data characteristics;

i- M: The dimension of the input features.

ii- Pt: The number of exemplars to be used in learning.

c) Initialization;

i- Number of iterations t=1.

ii- Number of committed category nodes C=1.

2) New input;

$$A_i^{(t)} = \begin{Bmatrix} a_i^{(t)} & for\ \ 1 \le i \le M \\ 1 - a_i^{(t)} & for\ \ M+1 \le i \le 2M \end{Bmatrix}$$

3) Compute the choice function for all committed category nodes;

$$T_j^{(t)} = \frac{\sum_{i=1}^{2M} \left(A_i^{(t)} \wedge w_{ij} \right)}{\alpha + \sum_{i=1}^{2M} w_{ij}} \quad ; j=1, ..., C$$

4) Reset: Determine the node J, which has the maximum choice value;

$$T_J^{(t)} = max\{T_j^{(t)}\} \quad ; j=1, ..., C$$

5) If $T_J^{(t)} = -1$ (all committed category nodes are in shut-off mode) then a new node (the node that its order in the category layer is C+1) should be committed;

i- Increase the number of committed nodes by one;

C=C+1

ii- If in fast-learning mode β=1;

Assign the values of the input feature to the weights of this node;

$$w_{iC}^{first} = A_i^{(t)} \quad ; i=1, ..., 2M$$

Else (normal mode)

$$w_{iC}^{first} = \beta A_i^{(t)} + (1-\beta) \qquad ;i=1, ..., 2M$$

iii- GOTO STEP (2)

6) Matching criterion: If ($\sum_{i=1}^{2M} (A_i^{(t)} \wedge w_{iJ}) / M < \rho$) then;

i- Shut-off this node to put it out of competition;

$$T_J^{(t)} = -1$$

ii- GOTO STEP (4)

7) Learning;

$$w_{iJ}^{new} = \beta(A_i^{(t)} \wedge w_{iJ}^{old}) + (1-\beta) w_{iJ}^{old}$$

8) If (t<Pt) then;

i- t=t+1

ii- GOTO STEP (2)

9) Training has been done. The network is ready for categorization.

3 Categorization of Flagged fuzzy ART and Compact Fuzzy ART

At the end of the training phase, all weights are fixed at their final update. The number of category node C is known. The network is ready for categorization.

While the mach value for original fuzzy ART is:

$$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{\sum_{i=1}^{2M} A_i}$$

It is for Flagged and Compact fuzzy ART:

$$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{M}$$

So, the training algorithm for both Flagged and Compact fuzzy ART is:

1) Input:

$$A_i^{(t)} = \begin{Bmatrix} a_i^{(t)} & for\ \ 1 \le i \le M \\ 1 - a_i^{(t)} & for\ \ M+1 \le i \le 2M \end{Bmatrix}$$

2) Compute the choice values for all committed nodes;

$$T_j^{(t)} = \frac{\sum_{i=1}^{2M}\left(A_i^{(t)} \wedge w_{ij}\right)}{\alpha + \sum_{i=1}^{2M} w_{ij}} \qquad ; j=1, \ldots, C$$

3) Determine the node *J*, which has the maximum choice function among all committed category nodes;

$$T_J^{(t)} = max\,\{T_j^{(t)}\} \quad ; j=1, \ldots, C$$

Match testing:

If (the match value for the winning node J: $\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{M} \ge \rho$) then;

Category node *J* represents the category of this input

Else

The network fails to categorize this input

5) If more categorization is needed GOTO STEP (1).

6) Categorization has been done.

4 Conclusions:

The comparison among the original Fuzzy ART, *Flagged*-Fuzzy ART, and *Compact*-Fuzzy ART is shown in (table 1). It shows clearly that *Flagged*-Fuzzy ART and *Compact*-Fuzzy ART are faster than the original algorithm of Fuzzy ART.

The main point that is influencing the reduction of the training time is the number of comparisons that are needed to determine the winning category node. They are N-1, C, and C-1 for the original Fuzzy ART, *Flagged*-Fuzzy ART and *Compact*-Fuzzy ART, respectively. More than that, since C increases from 1 up to its final value C_{final} at the end of training phase, an average of $C_{final}/2$ comparisons for Compact fuzzy ART compare to N comparisons for the fuzzy ART is require for determining the maximum choice value node. Keep in mind that we repeat the process of determination the maximum choice value node C times in case new category node must be committed to represent an input.

In addition to that the match values as well as the weights updating for the newly committed category node are requiring less computation. Moreover, the initialization for weights and choice values for category nodes is eliminated.

BIBLIOGRAPHY

Carpenter, G. A., and Grossberg, S.: A massively parallel architecture for a self-organizing neural pattern recognition machine, *Computer Vision, Graphic, and Image Processing*, (37) (1987), 54-115.

Carpenter, G. A., Grossberg, S., and Rosen, D. B.: Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system, *Neural Networks, (*4) (1991), 759-771.

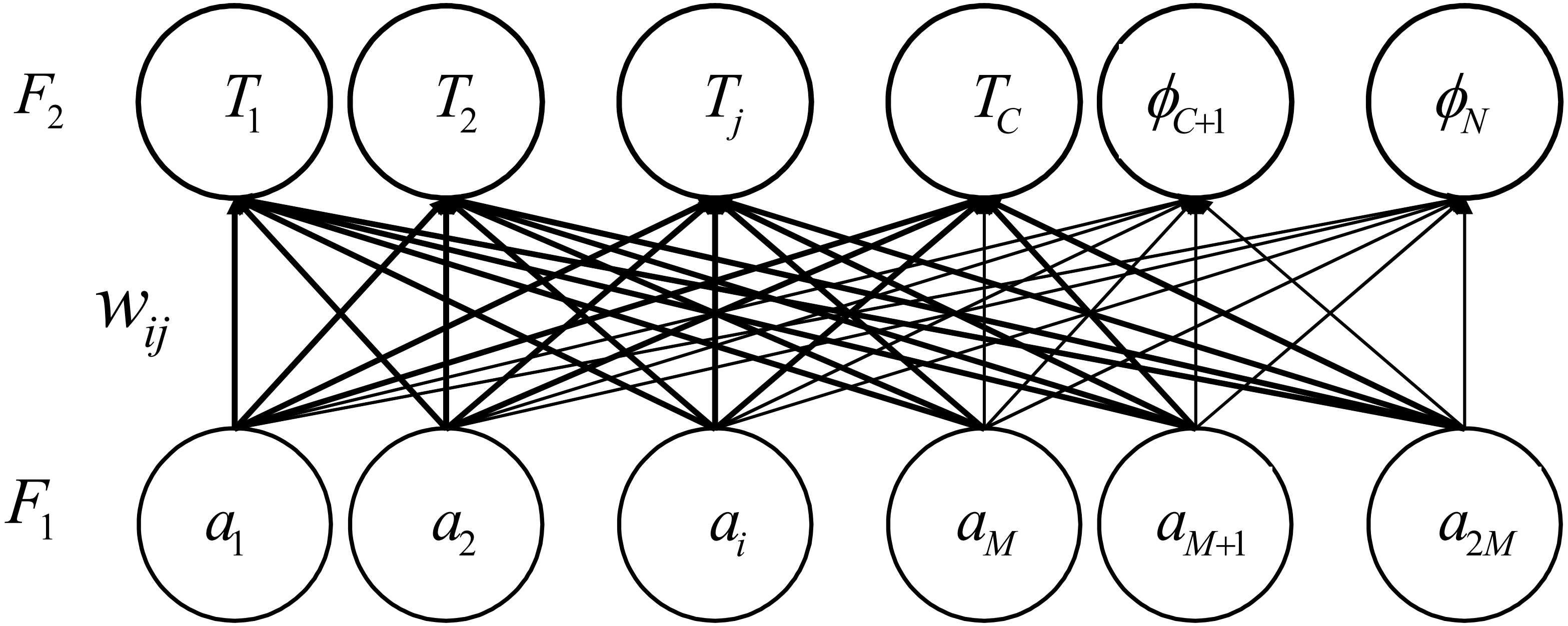

Figure 1: The architecture of Fuzzy ART. The full capacity N of the category layer is involved for determination the maximum choice value node J. They are shown in dark. Weights are connected to all category nodes. Weights that are connected to uncommitted category nodes are shown in light. This is because they are not learned yet.

The number of comparison which is needed to determine the maximum choice value node J is N-1, since it is carried out among all category nodes. This increases training time. If $J>C$ then the uncommitted category node with index C+1 has been committed, since its choice value $\phi_{C+1} = max,\{\phi_j)$; $j=C+1, ..., N$. That because these constant are arranged as $\phi_{C+1} < ... < \phi_N$. It has been prearranged this way to let category nodes to be committed in order to prevent the fragmentation of the category layer.

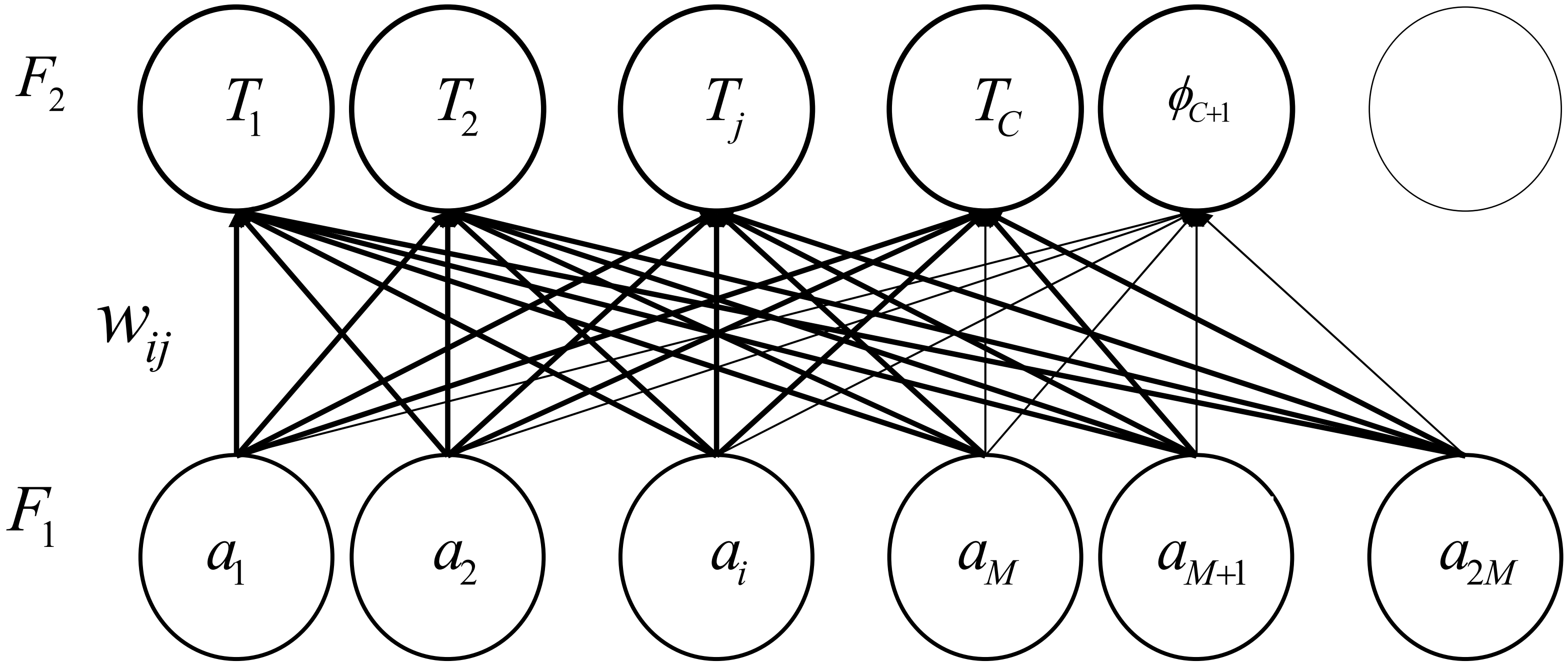

Figure-2: The architecture of *Flagged Fuzzy ART*. Only committed category nodes and the uncommitted category node with index *C*+1 in the category layer are involved in determination the maximum choice value node *J*. These category nodes are shown in dark. Weights are connected to all these category nodes. Category nodes that are not involved in determination the maximum choice value node, are shown in light. Weights are not connected to them. Weights that connected to the flagged node (uncommitted category node with index *C*+1) are shown in light. This is because they are not initiated yet. It will be initiated ($w_{i,C+1} = 1$; i=1, ..., *2M*)

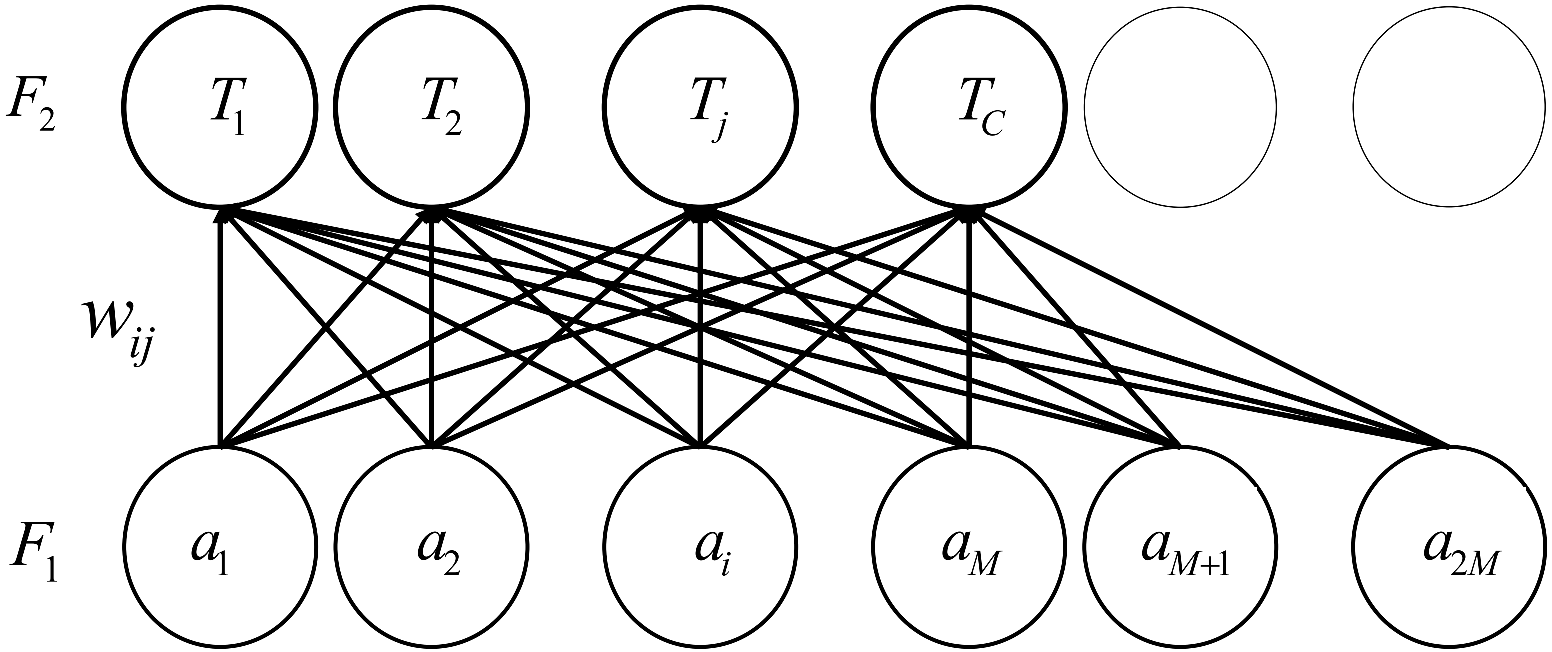

Figure-3 The architecture of *Compact Fuzzy ART*. Only committed category nodes are involved in determination the maximum choice value node *J*. These category nodes are shown in dark. Weights connect all input layer nodes to committed category nodes only. Uncommitted category nodes are shown in light. Weights are not connected to them since they are not committed yet (they are not assigned weights yet).

The number of comparison which is needed to determine the maximum choice value node is *C*-1, since it is carried out among committed category nodes only. This reduces training time.

	Fuzzy ART	**Flagged Fuzzy ART**	**Compact Fuzzy ART**
Initialization for Choice value	$0 < \phi_N < \ldots < \phi_j < \ldots < \phi_1 \cong 0$	$\phi_{C+1} = -0.1$	*None*
Compute choice function T_j	$T_j = \frac{\sum_{i=1}^{2M}(A_i \wedge w_{ij})}{\alpha + \sum_{i=1}^{2M} w_{ij}}; j = 1,\ldots,C$	*Same*	*Same*
Determination of T_{max}	$T_J = \max\{T_j; j = 1,\ldots,N\}$	$T_J = \max\{T_j; j = 1,\ldots,C+1\}$	$T_J = \max\{T_j; j = 1,\ldots,C\}$
Check for new committed node	$J > C$	$J > C$	$T_J = -1$
Number of comparison for T_{max}	$N - 1$	C	$C - 1$
Match testing	$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{\sum_{i=1}^{2M} A_i} \geq \rho$	$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{M} \geq \rho$	$\frac{\sum_{i=1}^{2M}(A_i \wedge w_{iJ})}{M} \geq \rho$
Weights initialization	$w_{ij} = 1; i = 1,\ldots,2M; j = 1,\ldots,N$	*None*	*None*
Weights updating for old node	$w_{iJ}^{new} = \beta(A_i \wedge w_{iJ}^{old}) + (1-\beta)w_{iJ}^{old}$	*Same*	*Same*
Weights updating for new node	$w_{iJ}^{new} = \beta(A_i \wedge w_{iJ}^{old}) + (1-\beta)w_{iJ}^{old}$	$w_{iC}^{first} = \beta * A_i + (1-\beta)$	$w_{iC}^{first} = \beta * A_i + (1-\beta)$

Table-1: Comparison among Original, Flagged, and Compact algorithms of Fuzzy ART. The last two have been developed in this study. Flagged and Compact algorithms are faster, however, Compact algorithm is recommended.