Fuzzy Cognitive Maps Based Models for Pattern Classification: Advances and Challenges

Gonzalo Nápoles, Maikel Leon Espinosa, Isel Grau, Koen Vanhoof and Rafael Bello

Abstract Fuzzy Cognitive Maps (FCMs) have proven to be a suitable methodology for the design of knowledge-based systems. By combining both uncertainty depiction and cognitive mapping, this technique represents the knowledge of systems that are characterized by ambiguity and complexity. In short, FCMs can be defined as recurrent neural networks that include elements of fuzzy logic during the knowledge engineering phase. While the literature contains many studies claiming how this Soft Computing technique is able to model complex and dynamical systems, we explore another promising research field: *the use of FCMs in solving pattern classification problems*. This is motivated by the transparency of the decision model attached to these cognitive, neural networks. In this chapter, we revise some prominent advances in the area of FCM-based classifiers and open challenges to be confronted.

1 Introduction

In the last years, *Fuzzy Cognitive Maps* (FCMs) [12] have notably increased their popularity within the scientific community. They constitute a suitable tool for the designing of knowledge-based systems, where one of the most relevant characteristics is the interpretability of the network topology. Not many computer science techniques can claim this valuable feature.

From the structural perspective, an FCM can be defined as a fuzzy digraph that describes the underlying behavior of an intelligent system in terms of concepts

G. Nápoles (✉) · K. Vanhoof
Hasselt Universiteit, Agoralaan gebouw D, Diepenbeek, Belgium
e-mail: gonzalo.napoles@uhasselt.be

M. Leon Espinosa
University of Miami, 5250 University Dr, Miami, FL, USA
e-mail: mleon@bus.miami.edu

I. Grau · R. Bello
Central University of Las Villas, Carretera Camajuaní km 5.5, Santa Clara, Cuba
e-mail: rbellop@uclv.edu.cu

D.A. Pelta and C. Cruz Corona (eds.), *Soft Computing Based Optimization and Decision Models*, Studies in Fuzziness and Soft Computing 360,
DOI 10.1007/978-3-319-64286-4_5

(i.e., objects, states, variables or entities). Such concepts comprise a precise meaning for the problem domain under analysis and they are connected by signed and weighted edges that denote causal relationships.

The sign and intensity of causal relations involve the quantification of a fuzzy linguistic variable that can be assigned by experts during a knowledge acquisition phase [13]. These elements recurrently interact when updating the activation value of each concept (or simply neuron). In point of fact, an FCM exploits an activation (state) vector by using a rule similar to the standard McCulloch-Pitts scheme [15]. Therefore, the activation value of each neuron is given by the value of the transformed weighted sum that this processing unit receives from connected neurons on the causal network. This activation value actually comprises an interpretable feature for the physical system under investigation. More explicitly, the higher the activation value of a neuron, the stronger its influence (positive or negative) over the connected neural entities. Of course, this influence also depends on the intensity of the causal relations connecting the actual neuron with the other neural processing entities.

FCM-based models can be understood as a kind of *recurrent neural networks* that support backward connections that sometimes form cycles in the causal graph. These backward relations (called feedback) enable the network to handle memory to compute the outputs of the current state and maintain a sort of recurrence to the past processing [6]. During the inference phase, the updating rule is repeated until the system converges to a fixed-point attractor or a maximal number of iterations is reached. The former implies that a hidden pattern was discovered [12] while the latter suggests that the outputs are cyclic or completely chaotic. Whichever the observed behavior, the recurrent network will produce a response (i.e., state vector) at each discrete-time step, which comprises the activation degree of all neurons of the model.

Although FCMs inherited many aspects from other neural models (i.e., the reasoning rule), there are some important differences regarding to other types of Artificial Neural Network (ANNs). Classical ANN models regularly perform like *black-boxes*, where both the neurons and the connections do not have a clear meaning for the problem itself, or results cannot easily be explained by the same predicting model. However, all neurons in an FCM have a precise meaning for the physical system being modeled and correspond to specific variables that form part of the solution. It should be highlighted that an FCM does not comprise hidden neurons since these entities could not be interpreted nor help at explaining why a solution is suitable for a given problem. If this were the case, the model becomes unfriendly for many further phases.

In the last years, FCMs have been widely studied due to its advantageous characteristics for handling complex systems. Less attention has been given to the development of FCM-based classifiers. *Pattern classification* [4] is one of the most ubiquitous real-world problems and certainly one at which humans really excel. It consists of identifying the right category (among those in a predefined set) to which an observed pattern belongs. These patterns are often described by a set of predictive attributes of numerical and/or nominal nature called features. Some successful classifiers include: artificial neural networks [7], support vector machines [8] or random forest [2]. Regrettably none of these black-box classifiers provides an inherent introspection into the reasoning process associated to the decision model. However, in some areas where machine learning models are applied, the transparency in their predictions is crucial.

Aiming at developing a novel classification model, Papakostas et al. [31, 32] introduced the notion of *FCM-based classifier*. The most prominent challenge to be confronted when constructing an FCM-based classifier relies on the approach to connect input and output neurons. It should be remarked that the topology of an FCM-based classifier must comprise a coherent and precise meaning for the physical system under investigation. This suggests that the intervention of human experts to define the network topology is usually required.

The development of accurate learning algorithms for computing the required parameters is another issue that deserves attention. In the literature, several unsupervised and supervised learning methods have been recently proposed [29]. These algorithms are mostly focused on computing the weight matrix that define the semantic of the whole cognitive system. However, the prediction capability of an FCM-based classifier does not exclusively depend on the weight set. Other aspects such as the network's capability the represent the problem domain or the convergence issues are equally important.

In this chapter, we focus on main advances on FCM-based classification and challenges that remain open problems for the scientific community. The rest of the manuscript is structured as follows. Section 2 briefly surveys theoretical aspects related to FCMs. Section 3 discusses about the transparency and usability of models for understanding the decision process. Section 4 describes the use of FCMs in the context of pattern classification. Section 5 describes the FCM-based models where input neurons denote information granules rather low-level features. To conclude, Sects. 6 and 7 will wrap-up the paper and highlight the main points of view of this proposal.

2 Fuzzy Cognitive Maps

FCMs can be seen as recurrent neural networks with interpretability features that have been widely used in modeling tasks [11]. They consist of a set of neural processing entities called concepts (neurons) and their causal relations. The activation value of such neurons regularly takes values in the [0, 1] interval, so the stronger the activation value of a neuron, the greater its impact on the network. Of course, connected weights are also relevant in this scheme. The strength of the causal relation between two neurons C_i and C_j is quantified by a numerical weight $w_{ij} \in [-1, 1]$ and denoted via a causal edge from C_i to C_j.

There are three possible types of causal relationships between neural processing units in an FCM-based network that express the type of influence from one neuron to the other, which are detailed as follows:

- If $w_{ij} > 0$ then an increase (decrement) in the cause C_i produces an increment (decrement) of the effect C_j with intensity $|w_{ij}|$.
- If $w_{ij} < 0$ then an increase (decrement) in the cause C_i produces an decrement (increment) of the neuron C_j with intensity $|w_{ij}|$.
- If $w_{ij} = 0$ then there is no causal relation between C_i and C_j.

Equation 1 shows the Kosko's activation rule, with $A^{(0)}$ being the initial state. This rule is iteratively repeated until a stopping condition is met. A new activation vector is calculated at each step t and after a fixed number of iterations, the FCM will be at one of the following states: (i) equilibrium point, (ii) limited cycle or (iii) chaotic behavior [12]. The FCM is said to have converged if it reaches a fixed-point attractor, otherwise the updating process terminates after a maximum number of iterations T is reached.

$$A_i^{(t+1)} = f\left(\sum_{j=1}^{M} w_{ji}A_j^{(t)}\right), i \neq j \tag{1}$$

The function $f(\cdot)$ in Eq. 1 denotes a monotonically non-decreasing nonlinear function used to clamp the activation value of each neuron to the allowed interval. Examples of such functions are the bivalent function, the trivalent function, and the sigmoid variants [37].

We put emphasis in the sigmoid function since it has exhibited superior prediction capabilities [3]. Equation 2 formalizes the non-linear transfer function used in our conducted researches, where λ is the sigmoid slope and h denotes the offset. Several studies reported at [1, 10, 14, 17, 27] have shown that such parameters are closely related with the network convergence.

$$f(A_i) = \frac{1}{1 + e^{-\lambda(A_i - h)}} \tag{2}$$

Equation 1 shows an inference rule widely used in many FCM-based applications, but it is not the only one. Stylios and Groumpos [36] proposed a modified inference rule, found at Eq. 3, where neurons take into account its own past value. This rule is preferred when updating the activation value of neurons that are not influenced by other neural processing entities.

$$A_i^{(t+1)} = f\left(\sum_{j=1}^{M} w_{ji}A_j^{(t)} + A_i^{(t)}\right), i \neq j \tag{3}$$

Another modified updating rule was proposed in [28] to avoid the conflicts emerging in the case of non-active neurons. Being more explicit, the rescaled inference depicted in Eq. 4 allows dealing with the scenarios where there is not information about an initial neuron-state and helps preventing the saturation problem. The reader can notice that we can obtain a similar effect by using a shifted sigmoid function with the adequate slope.

$$A_i^{(t+1)} = f\left(\sum_{j=1}^{M} w_{ji}(2A_j^{(t)} - 1) + (2A_i^{(t)} - 1)\right), i \neq j \tag{4}$$

If the cognitive network is able to converge, then the system will produce the same output towards the end, and therefore the activation degree of neurons will remain unaltered (or subject to infinitesimal changes). On the other hand, a cyclic FCM produces dissimilar responses with the exception of a few states that are periodically produced. The last possible scenario is related to chaotic configurations in which the network yields different state vectors. Formally, such situations are mathematically defined as follows:

- **Fixed-point** ($\exists t_\alpha \in \{1, 2, \dots, (T-1)\} : A^{(t+1)} = A^{(t)}, \forall t \geq t_\alpha$): the FCM produces the same state vector after the t_α-th iteration-step. This suggests that $A^{(t_\alpha)} = A^{(t_\alpha+1)} = A^{(t_\alpha+2)} = \cdots = A^{(t)}$.
- **Limit cycle** ($\exists t_\alpha, P \in \{1, 2, \dots, (T-1)\} : A^{(t+P)} = A^{(t)}, \forall t \geq t_\alpha$): the FCM periodically produces the same state vector after the t_α-th iteration-step. This suggests that $A^{(t_\alpha)} = A^{(t_\alpha+P)} = A^{(t_\alpha+2P)} = \cdots = A^{(t_\alpha+jP)}$ where $t_\alpha + jP \leq T$, such that $j \in \{1, 2, \dots, (T-1)\}$.
- **Chaos**: the FCM continues producing different state vectors for successive cycles, thus being impossible to make suitable decisions.

If the FCM is unable to converge, then the model will produce confusing responses and thus a pattern cannot be derived [26], thus being impossible to arrive at suitable conclusions. In presence of chaos or cyclic situations, the reasoning rule stops once a maximal number of iterations T is reached. In classification scenarios, the decision class is then calculated from the last cycle, but this output is partially unreliable due to the lack of convergence.

3 The Reasoning Process and Its Explainability

The *classification problem* [4] is about building a mapping $f : \mathcal{U} \rightarrow \mathcal{D}$ that assigns to each instance $x \in \mathcal{U}$ described by the attribute set $\boldsymbol{\Phi} = \{\phi_1, \dots, \phi_M\}$ a decision class D from the N possible ones in $\mathcal{D} = \{D_1, \dots, D_N\}$. The mapping is often learned in a *supervised* fashion, i.e., by relying on an existing set of previously labeled examples used to train the model. The learning process is regularly driven by the minimization of a cost/error function.

Researchers in *Machine Learning* are primarily focused on prediction rates. Regrettably, most accurate classifiers do not provide any mechanism to explain how they arrived at a particular conclusion and therefore behave like a “black-box”. Some classifiers like *Artificial Neural Networks*, *Support Vector Machines*, *Ensemble techniques* or *Random Forests* are well-known to be the most likely successful algorithms for addressing classification problems in terms of prediction rates. However, their lack of transparency negatively effects their usability in scenarios where understanding the decision process is required.

For example, neural computation is a widely studied research field within Artificial Intelligence. The main limitation of Artificial Neural Networks is their lack of transparency, which means that the network cannot justify its complex reasoning process. As a result, these models do not allow interpreting the semantic behind the physical system under investigation since the transparency is a necessary condition to build interpretable classifiers.

Aiming at elucidating the hidden reasoning process of black-boxes, several post-hoc procedures have been proposed. For example, one of these explanatory techniques used explicit IF-THEN rules for extracting knowledge from black-box classifiers while more recent procedures use symbolic approaches to approximate the model [9]. But whether such explanation is truly comprehensive and meaningful in the case of large trees or rule sets is questionable.

The transparency inherent to FCMs and their underlying neural foundations have motivated researchers to build interpretable FCM-based classifiers. In these models, the interpretability may be achieved through causal relations between neural entities defining the modeled system. Regrettably, building *accurate*, *truly interpretable* FCM-based classifiers involves difficult challenges.

4 Low-Level FCM-Based Classifiers

As already mentioned, FCMs have been widely studied due to their appealing properties for handling complex and dynamic systems, but the development of FCM-based classifiers has received less attention.

One of the firsts attempt to use FCMs in the context of pattern classification was implemented in [31, 32]. In these references, the authors defined the notion of *FCM-based classifiers* and proposed some generic architectures. The most prominent challenge to be faced when constructing an FCM-based classifier lies on how to connect input and output neurons.

It should be remarked that an FCM classifier's topology (i.e., concepts and causal relations) must comprise a coherent and precise meaning for the physical system being modeled. If the input neurons represent features of the classification problem, then we are in presence of a *low-level cognitive classifier* where neural processing units can be categorized as shown below:

Definition 1 We say that a neural processing entity C_i is an *independent input neuron* if its activation value does not depend on the other input neurons.

Definition 2 We say that a neural processing entity C_i is a *dependent input neuron* if its activation value is influenced by other connected neurons.

Definition 3 We say that a neural processing entity C_i in an FCM-based classifier is an *output neuron* if we can predict a decision class from its final activation value, which only depends on the connected input neurons.

Typically, independent and dependent input neurons are used to activate the cognitive networks since they often denote problem features. Output neurons, on the other hand, are used to compute the decision class for an initial activation vector. In the case of independent input neurons, they can propagate their initial activation vector and they are not influenced by any other input neurons, therefore their activation values remain static. Notice that the expert must ideally determine the role of each neurons and the way that input neurons are connected each other. In spite of this fact, Papakostas et al. [30] investigated three generic architectures for mapping the decision classes:

- **Class-per-output architecture**. Each decision class is mapped to an output neurons. Therefore, the predicted decision class corresponds to the label of the output neuron having the highest activation value.
- **Single-output architecture**. Each decision class is enclosed into the activation space of a single output neuron.

1. *Using a clustering approach*. Each class is associated with a cluster center. In the testing phase, the center having the closest distance to the projected activation value is assigned to the input instance.
2. *Using a thresholding approach*. Each decision class is associated with a pair of thresholds. In the testing phase, the interval comprising the projected activation value is then assigned to the input instance.

In these architectures, the human intervention is required during the construction stage, and even so, the supervised learning methods will probably fail in producing *authentic causal relations* since they just fit the model to the existing data. Therefore, we are losing the interpretability features attached to the network, although the decision process remains transparent.

On the other hand, the absence of hidden neural entities in these recurrent neural networks may probably lead to poor prediction rates. Aiming at boosting the prediction capability of FCM-based classifier, in [32] the authors put forth two hybrid typologies. Figures 1 and 2 show these typologies that include a *black-box* classifier to improve the overall prediction rates.

In the first model, the black-box produces a confidence degree per decision class. Sequentially, this vector is used as initial configuration for the FCM model that corrects the responses produced by the black-box. In the second model, the input neurons are also connected to output ones, so the predictions computed by the black-box classifier can be understood as a bias.

These hybrid models completely destroy the transparency attached to the cognitive network. If this happens, then, there is no real reason to use FCMs in classification scenarios, instead we may adopt black-box models such as Support Vector Machines, Neural Networks or Random Forests.

Another key element towards designing a low-level FCM-based classifier relies on the learning algorithm. The chief objective behind FCM learning has been to

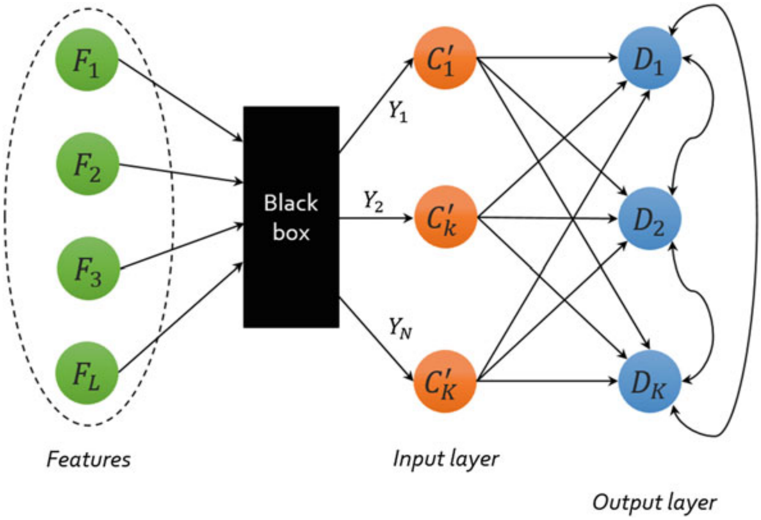

Fig. 1 Hybrid FCM-based classifier type-1

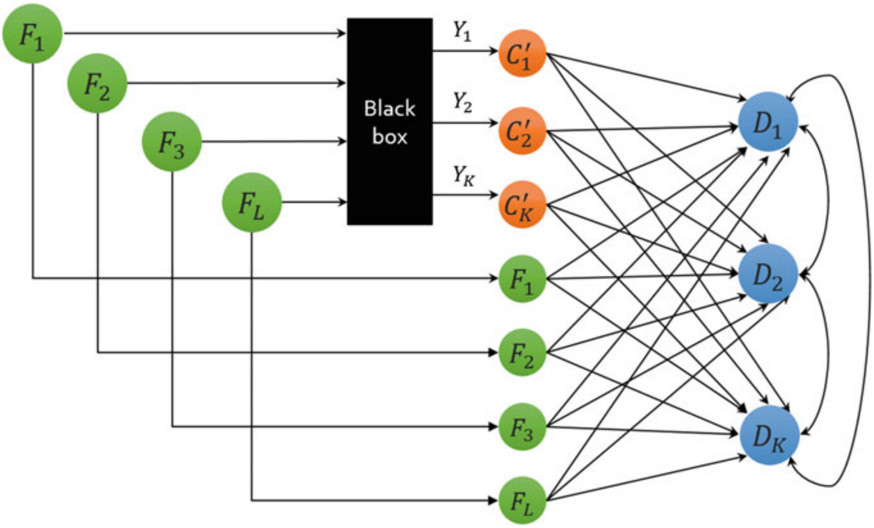

Fig. 2 Hybrid FCM-based classifier type-2

derive the weight matrix $W_{(M \times M)}$ that minimizes the prediction error based on expert intervention, available historical data or both. According to the their classification scheme, existing learning algorithms can be roughly gathered into two large groups: *unsupervised* and *supervised.*

4.1 Unsupervised Learning Algorithms

Hebbian-based learning methods are *unsupervised* procedures that do not require a set of labeled historical data, i.e., data in which the value of the decision feature(s) are previously known. The aim of learning FCMs by using adaptive Hebbian-based methods is to yield weight matrices on the basis of experts' knowledge and to improve the accuracy of previously set weights.

Papakostas et al. [30] thoroughly tested the performance of several Hebbian-type algorithms in classification scenarios, and concluded that these learning procedures regularly produce very poor classification rates.

More explicitly, Hebbian-type methods are convenient to fine-tune the weight set with a small deviation from the initial configuration. As a result, the adjusted causal relations partially preserve their physical meaning, which cannot be guaranteed when using a heuristic-based learning method. Of course, the requirement of experts' knowledge is a serious drawback. The flexibility on data requirements of these algorithms is the key aspect behind their poor generalization capability. This makes Hebbian-type algorithms unfit for solving pattern classification problems where multiple classes must be predicted.

4.2 Supervised Learning Algorithms

As an alternative to Hebbian-based methods, we can learn the network structure from data using heuristic-based algorithms [29] in a supervised fashion. Heuristic learning approaches aim at generating weight matrices that minimize an error function, viz. the difference between the expected responses and the map-inferred outputs. These methods are more expensive optimization techniques given that they regularly explore multiple candidate solutions. Besides, they require the definition of the objective function to be optimized, which is the core of these learning procedure, rather than the adopted search method.

Equation 5 formalizes an error function for pattern classification scenarios, where X denotes the weight matrix, K is the number of training instances, $\psi(.)$ is the decision model to be used for determining the class label, while S_k represents the expected decision class for the kth training instance. In the case of the single-output architecture, the class is computed from the activation value of the decision neuron at the last iteration-step.

$$E(X) = \frac{1}{k}\sum_{k=1}^{K}\begin{cases}\gamma_k, \ \psi(A_{Mk}^{(T)}) = S_k \\ 1, \ \psi(A_{Mk}^{(T)}) \neq S_k\end{cases} \tag{5}$$

Aiming at reducing the convergence error of the FCM-based classifier, the error function depicted in Eq. 5 uses a penalization factor γ_k for those instances that have been correctly classified. In short, the *convergence error* can be understood as the overall dissimilarity between the system response at each iteration, and the activation value at the last iteration-step.

Nápoles et al. [16, 17, 27] investigated the convergence of FCM-based classifiers and proposed a learning method to improve the system convergence, without altering the causal weights. More recently, they introduced an extended learning algorithm [26] where weights are estimated taking into account both accuracy and convergence. Based on these results, we propose a generalized measure to compute the convergence error of an FCM-based classifier.

Equation 6 shows the convergence error for the kth instance, where $\omega_t = t/T$ is the relevance of each iteration, M is the number of neurons, $N < M$ is the number of input-type ones, whereas $A_{ik}^{(t)}$ denotes the current activation value for the ith neuron. Moreover, π_k represents the centroid (ideal) point of the decision label associated to the kth training instance.

$$\gamma_k = \sum_{t=1}^{T}\frac{2\omega_t}{M(T+1)}\left(\sum_{i=1}^{N}\frac{(A_{ik}^{(t)} - A_{ik}^{(T)})^2}{N} + \sum_{i=1}^{M-N}\frac{(A_{ik}^{(t)} - \pi_k)^2}{M-N}\right) \tag{6}$$

Let us assume an FCM-based classifier using a single-output architecture and a thresholding approach, where the kth instance is associated with jth decision class. Equation 7 computes the centroid, where L_j^k and U_j^k denote the lower and upper decision thresholds, respectively.

$$\pi_k = \begin{cases} L_j^k, & L_j^k = 0 \\ U_j^k, & U_j^k = 1 \\ \frac{L_j^k + U_j^k}{2}, & L_j^k \neq 0, U_j^k \neq 1 \end{cases} \tag{7}$$

This approach introduces two key contributions in regard to the algorithm proposed in [26]. First, we remove the required parameters by measuring the convergence error if the target instance is correctly classified. This suggests that the system accuracy will always be favored. Second, we compute the converge error of sigmoid neurons according to their role in the network. The convergence error of input-type neurons is measured as the overall dissimilarity between the system response at each cycle, and the activation value at the last iteration. However, in the case of output-type neurons, we calculate the overall dissimilarity between each response, and the corresponding centroid.

Preliminary simulations using a Bioinformatic problem [21] have shown that this algorithms is capable of producing a suitable trade-off between convergence and accuracy. Ensuring the convergence helps in preventing the misclassifications of boundary instances, otherwise the model becomes fragile to perturbations. However, this algorithm cannot be generalized to other domains where the experts are unable to define the network topology.

5 High-Level Cognitive Classifiers

Cognitive mapping allows modeling different levels of interpretability, which depend on the abstraction degree. Neurons denoting entities with high abstraction level (i.e., information granules or prototypes) lead to high-level interpretable networks. If the level of abstraction is too high, then the physical system under investigation is difficult to analyze, so we are losing interpretability. On the other hand, defining attribute-level entities allow interpreting the system behavior at a low-level. However, sometimes the domain experts are unable to define precise, authentic causal relations with such specificity level.

High-level cognitive classifier refer to FCM-based models where input neurons denote information granules rather than low-level features. For example, Nápoles et al. [23, 24] introduced the notion of *rough cognitive mapping* in the context of pattern classification. The new classification model transforms the feature space into a granular one that is exploited using the neural inference rule present in FCM-based models. In these so-called *Rough Cognitive Networks* (RCNs), the weight matrix is automatically computed on the basis of the three-way decision rules [38] that construct three rough regions [33] to perform the classification process. The RCN model achieved competitive performance with respect to state-of-the-art methods in a real-world classification problems [23] as well as in a network intrusion detection scenario [22].

Figure 3 shows an RCN to solve any classification problem with two decision classes, where P_k, N_k and B_k are input neurons denoting the positive, negative and boundary regions related to the kth decision class.

More recently, two improved RCN models were introduced: *Rough Cognitive Ensembles* [20] and *Fuzzy-Rough Cognitive Networks* [25]. The purpose of these algorithms is to deal with the parametric requirement of rough cognitive classifiers while preserving their global prediction capabilities. The former is a granular ensemble model where each base RCN operates at a different granularity degree, whereas the latter replaced the crisp-rough constructs with fuzzy-rough ones. Numerical results have shown that both approaches are capable of outperforming the RCN algorithm. These modified algorithms perform comparably, thus we can achieve the same prediction rates using an ensemble composed of several networks that using a single fuzzy-rough classifier!

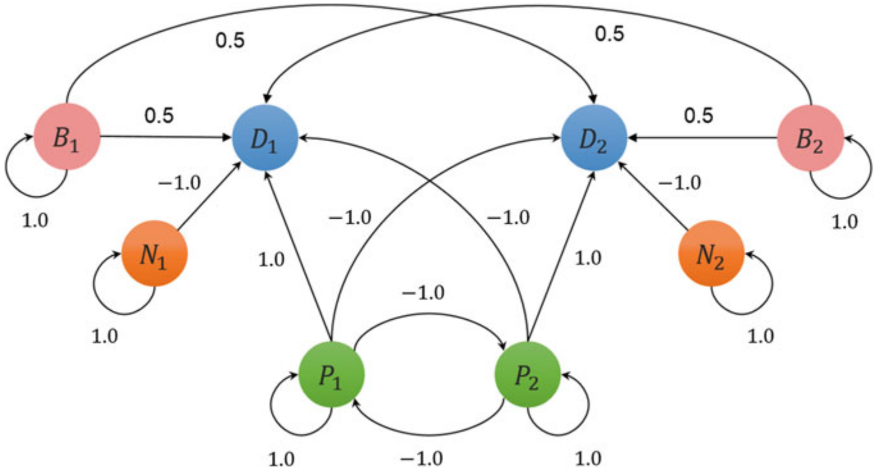

Fig. 3 RCN-based classifier for binary problems

Inspired on the RCN semantics and the approaches discussed in [34, 35], Nápoles et al. [19] proposed a *partitive granular cognitive map* to solve graded multi-label classification problems. In these machine learning problems, the goal is to predict the degree to which each instance relates to each available decision class. Three different FCM topologies were studied and several convergence features were included into the supervised learning methodology. Numerical experiments confirmed the ability of these granular classifiers to accurately estimate the degree of association between an object and each label.

It is worth highlighting the transparency on the decision model attached to Rough Cognitive Networks. In these models, we can interpret the physical system at a high-level by relying on the semantics behind the information constructs. However, a low-level reasoning is not possible, even when the classifier's decision process remains transparent and comprehensible.

6 Remaining Challenges

The development of accurate, truly interpretable FCM-based classifiers involves three main challenges, that still remain open:

- **Construction issues**. FCMs are knowledge-based techniques that regularly require the intervention of experts to define the network topology, i.e., the neurons and causal relations connecting them. Alternatively, we can learn the network structure from data using heuristic-based algorithms in a supervised fashion. However, these methods cannot produce *authentic causal relations* describing the system under analysis since they are oriented to fit the network to the historical data, without considering the system semantics. This implies that we cannot interpret the problem domain from such models, even if the FCM inference process is still transparent. Some authors attempt overcoming this drawback using correlation measures, which fail in capturing the underlying semantics behind causal relations. Being more explicit, it is well-known that causality does surely imply the existence of correlation, but the opposite does not necessarily hold.

- **Accuracy issues**. Generally speaking, the prediction rates of FCM-based classifiers are poor when compared with standard black-box models, mainly due to their limitations to represent the problem domain and the absence of theoretically sound learning algorithms. Papakostas et al. [30] concluded that Hebbian-based algorithms are not suitable in pattern classification environments, while the performance of heuristic-based learning methods quickly deteriorates when the number of neurons scales up. Froelich [5] proposed a promising post-optimization method to improve the prediction rates of FCM-based classifiers using a single-output architecture. Notice however that the overall prediction rates achieved by this method will heavily depend on the learning algorithm used to estimate the weight set.
- **Convergence issues**. FCM-based networks are recurrent cognitive systems that produce an output vector at each discrete-time step. This procedure is repeated until either the map converges or a maximal number of iterations is reached. Without ensuring the convergence, the model becomes unreliable and decision making becomes impossible. Regrettably, heuristic-based methods cannot ensure the FCM convergence, which implies that the resultant models are no longer interpretable and therefore, there is no reason to use cognitive mapping in pattern classification environments. More recently, Nápoles and his collaborators [16, 17, 26, 27] obtained promising results toward improving the convergence of FCM-based models without modifying the causal relations. However, analytical results reported in [18] have shown that establishing a suitable balance between convergence and accuracy cannot always be achieved without altering the weights.

It should be observed that the accuracy and convergence issues are mathematical challenges that can be present in other Machine Learning approaches. After all, the main purpose of traditional classifiers is to achieve the best possible prediction rates. The construction issues are, however, more delicate. Defining authentic causal relations between neural entities is the key aspect towards designing truly interpretable FCM-based systems. Otherwise, the model will produce misleading results when performing WHAT-IF simulations.

As far as we know, there is no learning method able to discover authentic causal structures from historical records due to the lack of well-established statistical tests for measuring causality. Even some authors affirm that the term “causality” is a philosophic concept that cannot possibly be measured in a numerical way without performing controlled experiments.

7 Conclusions

The use of FCMs for modeling real-life problems by recreating virtual scenarios have been demonstrated and reported in literature. These knowledge-based networks have been used as a modeling tool to analyze the behavior of complex systems, where it is very difficult to describe the entire system by a precise mathematical model. Consequently, it is easier and more practical to represent the decision-making process in a graphical way.

This paper explored the development of FCM-based classifiers and focused on the wide research avenues it provides. In spite of the detected shortcomings and challenges, the transparency inherent to cognitive mapping keeps motivating researchers to build interpretable FCM-based classifiers. In these models, the interpretability is achieved through causal relations between neurons defining the system under analysis. FCM-based models also provide other set of attractive characteristics: they are able to discover hidden patterns, are flexible, dynamic, combinable and tunable from different perspectives.

The FCM-based modeling approach allows building the network in presence of incomplete, conflicting or subjective information. Moreover, the inherent neural features of cognitive mapping provide a promising research avenue towards improving their accuracy in prediction scenarios. This suggests that FCM-based models could be as efficient as black-box models while retaining their ability to elucidate the system behavior through causal relations. Precisely, this conjecture, among other factors, keeps this research subject as a challenge open to the scientific community and a lively field of research.

Acknowledgements This work was partially supported by the Research Council of Hasselt University. Moreover, Gonzalo Nápoles would like to thank Frank Vanhoenshoven, Hasselt University, for his critical remarks and valuable suggestions.

References

1. Boutalis, Y., Kottas, T.L., Christodoulou, M.: Adaptive estimation of fuzzy cognitive maps with proven stability and parameter convergence. IEEE Trans. Fuzzy Syst. **17**(4), 874–889 (2009)
2. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
3. Bueno, S., Salmeron, J.L.: Benchmarking main activation functions in fuzzy cognitive maps. Expert Syst. Appl. **36**(3), 5221–5229 (2009)
4. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley (2012)
5. Froelich, W.: Towards improving the efficiency of the fuzzy cognitive map classifier. Neurocomputing **232**, 83–93 (2017)
6. Grau, I., Nápoles, G., Bonet, I., Garcia, M.M.: Backpropagation through time algorithm for training recurrent neural networks using variable length instances. Computación y Sistemas **17**(1), 15–24 (2013)
7. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice Hall PTR, Upper Saddle River (1998)

8. Hearst, M.A., Dumais, S.T., Osman, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intell. Syst. Appl. **13**(4), 18–28 (1998)
9. Jacobsson, H.: Rule extraction from recurrent neural networks: a taxonomy and review. Neural Comput. **17**(6), 1223–1263 (2005)
10. Knight, C.J., Lloyd, D.J., Penn, A.S.: Linear and sigmoidal fuzzy cognitive maps: an analysis of fixed points. Appl. Soft Comput. **15**, 193–202 (2014)
11. Kosko, B.: Fuzzy cognitive maps. Int. J. Man-Mach. Stud. **24**(1), 65–75 (1986)
12. Kosko, B.: Hidden patterns in combined and adaptive knowledge networks. Int. J. Approx. Reason. **2**(4), 377–393 (1988)
13. Kosko, B.: Fuzzy Engineering. Prentice Hall (1997)
14. Kottas, T.L., Boutalis, Y.S., Christodoulou, M.A.: Fuzzy cognitive networks: adaptive network estimation and control paradigms. In: Glykas, M. (ed.) Fuzzy Cognitive Maps: Advances in Theory, Methodologies, Tools and Applications, pp. 89–134. Springer, Berlin, Heidelberg (2010)
15. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. In: Anderson, J.A., Rosenfeld, E. (eds.) Neurocomputing: Foundations of Research, pp. 15–27. MIT Press, Cambridge (1988)
16. Nápoles, G., Bello, R., Vanhoof, K.: Learning Stability Features on Sigmoid Fuzzy Cognitive Maps through a Swarm Intelligence Approach, pp. 270–277. Springer, Berlin, Heidelberg (2013)
17. Nápoles, G., Bello, R., Vanhoof, K.: How to improve the convergence on sigmoid fuzzy cognitive maps? Intell. Data Anal. **18**(6S), S77–S88 (2014)
18. Nápoles, G., Concepción, L., Falcon, R., Bello, R., Vanhoof, K.: On the accuracy-convergence trade-off in sigmoid fuzzy cognitive maps. IEEE Trans. Fuzzy Syst. (submitted) (2017)
19. Nápoles, G., Falcon, R., Papageorgiou, E., Bello, R., Vanhoof, K.: Partitive granular cognitive maps to graded multilabel classification. In: 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1363–1370 (2016)
20. Nápoles, G., Falcon, R., Papageorgiou, E., Bello, R., Vanhoof, K.: Rough cognitive ensembles. Int. J. Approx. Reason. **85**, 79–96 (2017)
21. Nápoles, G., Grau, I., Bello, R., Grau, R.: Two-steps learning of fuzzy cognitive maps for prediction and knowledge discovery on the HIV-1 drug resistance. Expert Syst. Appl. **41**(3), 821–830 (2014)
22. Nápoles, G., Grau, I., Falcon, R., Bello, R., Vanhoof, K.: A granular intrusion detection system using rough cognitive networks. In: Abielmona, R., Falcon, R., Zincir-Heywood, N., Abbass, H. (eds.) Recent Advances in Computational Intelligence in Defense and Security, chapter 7. Springer (2016)
23. Nápoles, G., Grau, I., Papageorgiou, E., Bello, R., Vanhoof, K.: Rough cognitive networks. Knowl.-Based Syst. **91**, 46–61 (2016)
24. Nápoles, G., Grau, I., Vanhoof, K., Bello, R.: Hybrid model based on rough sets theory and fuzzy cognitive maps for decision-making. In: International Conference on Rough Sets and Intelligent Systems Paradigms, pp. 169–178. Springer (2014)
25. Nápoles, G., Mosquera, C., Falcon, R., Grau, I., Bello, R., Vanhoof, K.: Fuzzy-rough cognitive networks. Neural Netw. (2017)
26. Nápoles, G., Papageorgiou, E., Bello, R., Vanhoof, K.: Learning and convergence of fuzzy cognitive maps used in pattern recognition. Neural Process. Lett. 1–14 (2016)
27. Nápoles, G., Papageorgiou, E., Bello, R., Vanhoof, K.: On the convergence of sigmoid fuzzy cognitive maps. Inf. Sci. **349–350**, 154–171 (2016)
28. Papageorgiou, E.I.: A new methodology for decisions in medical informatics using fuzzy cognitive maps based on fuzzy rule-extraction techniques. Appl. Soft Comput. **11**(1), 500–513 (2011)
29. Papageorgiou, E.I.: Learning algorithms for fuzzy cognitive maps—a review study. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **42**(2), 150–163 (2012)
30. Papakostas, G., Koulouriotis, D., Polydoros, A., Tourassis, V.: Towards hebbian learning of fuzzy cognitive maps in pattern classification problems. Expert Syst. Appl. **39**(12), 10620–10629 (2012)

31. Papakostas, G.A., Boutalis, Y.S., Koulouriotis, D.E., Mertzios, B.G.: Fuzzy cognitive maps for pattern recognition applications. Int. J. Pattern Recogn. Artif. Intell. **22**(8), 1461–1486 (2008)
32. Papakostas, G.A., Koulouriotis, D.E.: Classifying patterns using fuzzy cognitive maps. In: Glykas, M. (ed.) Fuzzy Cognitive Maps: Advances in Theory, Methodologies, Tools and Applications, pp. 291–306. Springer, Berlin, Heidelberg (2010)
33. Pawlak, Z.: Rough sets. Int. J. Comput. Inf. Sci. **11**(5), 341–356 (1982)
34. Pedrycz, W.: The design of cognitive maps: a study in synergy of granular computing and evolutionary optimization. Expert Syst. Appl. **37**(10), 7288–7294 (2010)
35. Pedrycz, W., Homenda, W.: From fuzzy cognitive maps to granular cognitive maps. IEEE Trans. Fuzzy Syst. **22**(4), 859–869 (2014)
36. Stylios, C.D., Groumpos, P.P.: Modeling complex systems using fuzzy cognitive maps. IEEE Trans. Syst. Man Cybern.—Part A: Syst. Hum. **34**(1), 155–162 (2004)
37. Tsadiras, A.K.: Comparing the inference capabilities of binary, trivalent and sigmoid fuzzy cognitive maps. Inf. Sci. **178**(20), 3880–3894 (2008)
38. Yao, Y.: Three-way decisions with probabilistic rough sets. Inf. Sci. **180**(3), 341–353 (2010)