ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

Extracting fuzzy rules from "mental" images generated by modified WiSARD perceptrons

Bruno P. A. Grieco[1], Priscila M. V. Lima[1], Massimo De Gregorio[2] and Felipe M. G. França[1] *

1- Systems Engineering and Computer Science Program - COPPE,
Universidade Federal do Rio de Janeiro - Brazil
2- Istituto di Cibernetica "Eduardo Caianiello"- CNR
Via Campi Flegrei 34, 80078 Pozzuoli (NA) - Italy

Abstract. The pioneering WiSARD weightless neural classifier is based on the collective response of RAM-based neurons. The ability of producing prototypes, analog to "mental images", from learned categories, was first introduced in the DRASiW model. By counting the frequency of write accesses at each RAM neuron during the training phase, it is possible to associate the most accessed addresses to the corresponding input field contents that defined them. This work is about extracting information from such frequency counting in the form of fuzzy rules as an alternative way to describe the same mental images produced by DRASiW as logical prototypes.

1 Introduction

Proposed by Wilkes, Stonham, and Aleksander in 1984, the WiSARD (Wilkes, Stonham, Aleksander Recognition Device) perceptron became a pioneering representative in the field of Weightless Neural Networks [1]. WiSARD takes a string of bits (a bitmap) as input. This input is parsed into a set of uncorrelated *n*-tuples, each *n*-tuple may be regarded as a specific memory address and, this way, the input field is completely covered once.

Each of such kind of covering of the input field by RAM neurons defines a WiSARD *discriminator* that is assigned to the recognition of a target class/category. However, as it will be explained in the following section, this means that the WiSARD model is a unidirectional structure, a perceptron. The DRASiW model was introduced as a way of providing retro-classification capabilities to the WiSARD model in such a way one can ask for *prototypes* of already learnt categories [2], i.e., each discriminator is able to produce a representative example of a learnt class from trained patterns. In order to allow this, write accesses to RAM neurons were altered to provide a counting procedure that may be later reversed to the input field, where the "mental" image is produced, thus yielding a bidirectional structure.

*This work was partially supported by CNPq (Proc. 306070/2007-3) and FAPERJ (Proc. E-26/101.773/2008), Brazil.

This work presents a novel way of representing learnt categories, based on the same qualitative information of the DRASiW model, in the form of fuzzy rules. Unlike relevant related work where fuzzy logic improves weightless neural network performance on categorization and cognition [3][4], and where Boolean [5] and fuzzy rule [6] extraction are explored, the main goal of this paper is to show that it is possible to extract representative fuzzy rules of acquired knowledge via pattern exposure in the WiSARD/DRASiW weightless neural models.

2 The WiSARD Revisited

During training phase, each pattern of the training set (bitmap) is presented to the WiSARD; parts of the bitmap (*n*-tuples) are presented to RAM neurons as addresses. Each RAM position addressed is then set to (Boolean) "one" (1) – non-addressed positions are previously initialized with "zero" (0). During recognition phase, when a test pattern is presented to the WiSARD, each RAM neuron outputs the value of the accessed address by the test pattern, 0 or 1, meaning that the pattern was seen in the training phase or not, respectively.

A RAM neuron alone is not responsible for the pattern recognition. RAM neurons may not be very large in terms of the tuple arity, since a binary address of size n yields an addressable space of size 2^n. A set of RAM neurons, called *discriminator*, accumulates the output of all RAM neurons, producing a pertinence output. Figure 1 illustrates the arrangement. Whenever m classes of patterns are considered, m discriminators, in a localized representation, will be used. The amount of allocated memory would add up to $m \times 2^n$, where n is the size of each tuple.

Each discriminator is separately trained with each one of the m classes. When presented with the test pattern, each discriminator will output the number of neurons that fired accordingly. The discriminator with the highest output sum, therefore, should represent the presented pattern.

3 DRASiW

After training what has been learnt? One of the problems underlying the study of neural networks is the extraction of the learned rules (if any) from the training set. This is not a straightforward procedure [7]. Facing this problem in the realm of weightless neural networks may be eased by a small modification of the original WiSARD model. The DRASiW model [2] addresses the problem of "recalling"[†]

[†] The term "recall" will be used throughout this paper with the meaning of accessing "mental" images, while "recover" will be used with the meaning of the output provided by the system for a presented pattern.

learnt patterns, or "mental" images, fed to the modified WiSARD (DRASiW) during training.

The required modification is very simple: instead of representing trained pattern by solely setting a bit to one in a specific memory address, memory locations in the DRASiW model are counters that are incremented each time an input pattern addresses them. The "mental" image recall is produced based on the contents of such array of counters, containing how many times each input bit was presented to DRASiW. Frequently accessed positions are mapped to frequently used addresses produced from the input field.

Consider the following example, illustrated in Figure 1. A discriminator is made to recognize straight vertical lines from a 25 bit (5x5) black and white image – the WiSARD may be used to recognize any pattern that may be represented as a bitmap. Image representations are straightforwardly converted to bit streams just by concatenating their lines into a single lengthy one.

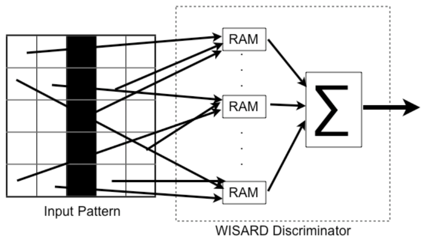
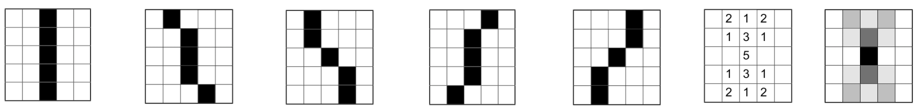


Fig. 1: A 5x5 WiSARD discriminator being trained with a vertical straight line.

The next example shows a 25bit bitmap with a discriminator that is composed by 6 four-bit RAM neurons plus one extra 1-bit RAM neuron to complete the 25th pixel. To train this discriminator, five training patterns were presented, each representing a separate instance of the class "vertical straight line", as follows: vertical straight line (Figure 2.a), slightly left tilted line (2.b), left tilted line (2.c), slightly right tilted line (2.d) and right tilted line (2.e). The input address shuffling as well as the training set presentation order are both irrelevant for this analysis.



(a) (b) (c) (d) (e) (f) (g)

Fig. 2 : Training patterns for the "vertical straight line" class.

The algorithm for recalling the output array checks each RAM neuron and for each of the input connections (tuple coordinate), sums all the counters accessed by that bit. This is the recalled value for that specific pixel. The output for the training set may be seen in Figure 2.f, it was formatted as a 5x5 matrix to present a more meaningful picture. This output may also be normalized and presented as a gray scale

ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

image (usually with pixels ranging from 0 to 255), as shown in Figure 2.g as a more human-friendly version. It is clearly noticeable that the center pixel has a value of 5, meaning that it is extremely relevant for this discriminator while the top center pixel has a value of 1, meaning that it is somewhat relevant but a side pixel with value of 0 means that it's not relevant at all. While not in recall mode the DRASiW recovery mode interprets the RAM neuron as zero meaning that the neuron will not fire and anything else meaning that the neuron will fire.

4 Fuzzy Rule Extraction

The term "mental" images apply very well to this procedure in order that it provides a simple way to communicate between the system and the user. However, not always the interface of the classification system will be made directly with human users. It is often desirable that the system may pipe its output to yet another automated system. This other system might be written in Prolog, or any other language that presumes a symbolic logical reasoning as its framework. In those cases, the recall must be achieved through a set of rules rather than a "mental" image.

A Fuzzy Logic paradigm was invoked in order to produce those rules. By analyzing each RAM individually, it is possible to notice that each entry may be regarded as rule stating that if input **a** equals 1 and input **b** equals 0, then this RAM fires an output. Since each entry has an associated value. If this value is normalized in regard to the highest output value, it may be said that it represents the membership of this entry within the set of firing rules for this RAM. The same may be said about the inputs themselves. If normalized accordingly, they produce a table that represents the membership value of that specific input on the RAM result. Since an input may be either 0 or 1, both values must be taken on account when creating the table. A representation of what an individual RAM might look is shown in Figure 3.a. Figure 3.b shows a normalized RAM. The bit input table is shown in Figure 3.c with absolute values and in Figure 3.d with normalized values.

With this paradigm as base, a procedure was derived in order to produce the expected fuzzy rules like the one shown as an example in figure 4.

0000	0
0001	0
0010	0
0011	3
...	...
1100	0
1101	2
1110	0
1111	0

a b c d

(a)

0000	0
0001	0
0010	0
0011	0,6
...	...
1100	0
1101	0,4
1110	0
1111	0

a b c d

(b)

a	2
~a	3
b	2
~b	3
c	3
~c	2
d	5
~d	0

(c)

a	0,4
~a	0,6
b	0,4
~b	0,6
c	0,6
~c	0,4
d	1
~d	0

(d)

Fig. 3: RAM Neuron rules and input bits tables, original and normalized

$$\{[((\neg p18\times0.8)\wedge(\neg p10\times1.0))\times0.2] \vee [((\neg p18\times0.8)\wedge(\neg p10\times1.0))\times0.4] \vee [((\neg p18\times0.8)\wedge(\neg p10\times1.0))\times0.2] \vee [((p18\times0.2)\wedge(\neg p10\times1.0))\times0.2]\}$$

Fig. 4: Example of expected extracted Fuzzy Expression

After exporting the generated fuzzy rules for the trained pattern, an experiment was conducted implementing the rules on a regular spreadsheet. The input was modeled as cells that could receive either true or false value; the bit input table was modeled so that a cell would represent the fuzzy membership function for each of the inputs. For the AND function, the MIN *t*-norm was used and, for the OR function, the MAX *t*-conorm was followed. Once chosen the membership value for the input (through the *t*-norm), it was multiplied by the membership value of the chosen rule (*t*-conorm). This way, each set of rules provided by a discriminator produces a fuzzy output value. Those results were then added as a WiSARD discriminator would.

5 Test Results

Following the experiment, a thorough statistical analysis was performed in order to compare the output of the DRASiW discriminator with the proposed Fuzzy discriminator. The input bitmap represented a 5x5 grid. A DRASiW that contained seven RAMs was trained with the five patterns shown in figure 2. The Fuzzy expression based on the training patterns was then extracted. All possible permutations on this bitmap (2^{25}) were generated and each permutation was presented to both the DRASiW and the Fuzzy discriminators, the result of both operations were stored for further comparison.

Comparisons were made on account of the number of patterns that were accepted or rejected by each discriminator. Figure 5 represents the ammount of samples that were filtered by each discriminator during the analysis. As one may observe, the smaller the discriminator output value, the smaller the number of patterns that are rejected by it.

For the DRASiW discriminator, the categories ranged from the seven possible outputs generated, that is, the number of RAMs that fired positive for each input. For the Fuzzy discriminator, seven categories were created, each one representing a possible output value, ranging from 0 to 3.5 with 0.5 increments. While the number of different generated possibilities is greater than seven this number was chosen for ease of histogram comparison. It should be noted that the histogram value represents the mean for each group i.e. category 3.0 represents values from 2.75 to 3.25. Since the maximum output value for this, specific, Fuzzy discriminator is 3.16, category 3.5 received no hits, but as noted before, it was left for comparison.

Analysis of the results shows that the Fuzzy discriminator is sharper than the DRASiW. Using a threshold value of 2.0 already crops the possible output to 0.04% of the whole possible set compared to 0.07% using the 5 threshold.

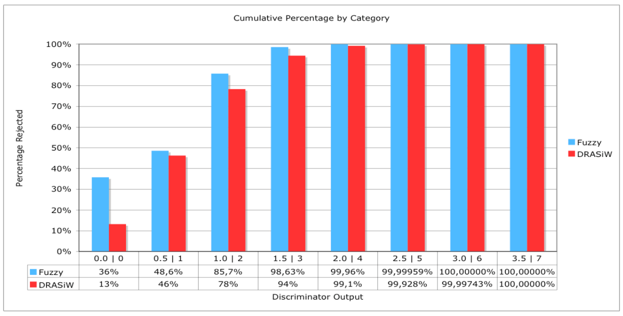


Fig. 5: Cumulative Percentage by Category

6 Conclusion

When compared with the output of an ordinary WiSARD, the results were consistent with each other. Patterns recognized by the WiSARD also received good ratings of the fuzzy model and vice-versa; patterns rejected by one also received poor ratings from the other. As a supplementary result, the fuzzy model appeared to be more sensitive. Patterns with same WiSARD output produced differentiated fuzzy ratings. This result strongly suggests that the fuzzy rules may be used as a quality advisor in the pattern recognition process.

References

[1] I. Aleksander, W.V. Thomas and P.A. Bowden, WiSARD: A Radical Step Forward in Image Recognition, *Sensor Review,* pp. 120-124, July 1984.

[2] C. M. Soares, C. L. F. da Silva, M. De Gregorio and F. M. G. França, Uma Implementação em Software do Classificador WiSARD. In A. P. Braga and T. B. Ludermir, ed., *V Simpósio Brasileiro de Redes Neurais* (SBRN 1998), vol. 2, pp. 225-229, December 1998 (in Portuguese).

[3] J. A. Tome and J. P. Carvalho, Fuzzy Boolean Networks Learning Behaviour, *Proc. of the 7th International Conference on Intelligent Systems Design and Applications* (ISDA 2007), pp. 889-894, October 2007.

[4] J. A. Tome and J. P. Carvalho, Decision validation and emotional layers on Fuzzy Boolean Networks, *Proc. of the IEEE Annual Meeting of the Fuzzy Information* (NAFIPS '04), vol.1, pp. 136-139, June 2004.

[5] T. B. Ludermir, M. C. P. de Souto and W. R. de Oliveira, Weightless Neural Netowrks: Knowledge-based Inference Systems, In M. M. B. R. Vellasco, M. C. P. de Souto and J. J. F. Cerqueira, ed., *Proc. of the 10th Brazilian Symposium on Neural Networks* (SBRN 2008), pp. 207-212, Nov. 2008.

[6] R. K. Brouwer, Fuzzy rule extraction from a feed forward neural network by training a representative fuzzy neural network using gradient descent, *Proc. of the 2004 IEEE International Conference on Industrial Technology* (IEEE ICIT '04), vol.3, pp. 1168-1172, December 2004.

[7] E. Burattini, P. Coraggio, M. De Gregorio, and M. Staffa, Agent WiSARD in a 3D World. In J. Mira and J. R. Álvarez, ed., proc. of the *1st International Work-Conference on the Interplay Between Natural and Artificial Computation* (IWINAC 2005), LNCS 3562, pp. 272-280, Springer-Verlag, 2005.