ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

“Mental Images” driven classification

Gianluca Coda, Massimo De Gregorio, Antonio Sorgente and Paolo Vanacore

Istituto di Scienze Applicate e Sistemi Intelligenti – CNR, Italy

Abstract. Common sense rules are a form of implicit knowledge acquired through experience and observation of the world around us, and used by both humans and machines to reason and to make decisions about the surrounding environment. Artificial Intelligence systems can extract these rules by mining data and apply them to many predictive tasks. Herein, we first present a new method for extracting rules from DRASiW “Mental Images” (MI) and then how to exploit them to improve the classification performance of the system. The latter is confirmed by the obtained results.

1 Introduction

Neural networks rule extraction is a field of study that aims to make system decisions interpretable [1][2] and comprehensible, tackling the problem of black-box models. The rules provide explanations [3] for the predictions made and help in understanding the decision-making process of the network. This is crucial in applications such as industrial control, medical diagnostics, and financial forecasting, where decisions need to be clearly justified and explained [4, 5].

In the past, two approaches have already been proposed for extracting rules from MI*s*. In [6], the authors proposed a procedure to convert the MI*s* information content into a set of fuzzy rules and compared its classification accuracy with that of DRASiW. A further approach, which is similar but different from the one proposed in this work, was introduced in [7]. Both approaches [6][7] are based on the analysis of sub-pattern frequency but in [7] this is carried out by comparing the same RAM memory contents among different discriminators. In this work, the extraction of rules is done by analysing the whole MI all together.

Making explicit and available the knowledge that is not expressed or caught by the training set is the aim of this work. Herein, we propose a procedure for extracting rules from DRASiW MI*s* with the purpose of using them to improve the system performance. After the training phase, a set of rules (if any) are extracted and used as a filter between the input and the discriminators.

2 The DRASiW Classifier

WiSARD (Wilkie, Stonham and Aleksander’s Recognition Device) [8] belongs to the class of Weightless Neural Networks (WNNs) [9], and it is based on a neural model which uses lookup tables to store the function computed by each neuron rather than storing it in weights of neuron connections. A WiSARD is composed of a set of classifiers, called discriminators, each one trained by binary patterns belonging to a particular category/class. Therefore, a WiSARD has as many discriminators as the number of categories/classes it should be able to

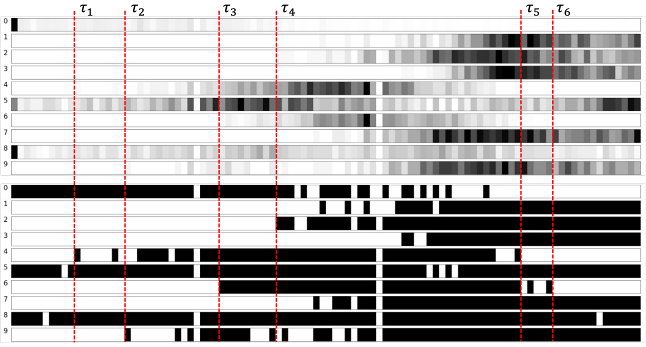

Fig. 1: MI*s* of the 10 *Optdigits* classes for the feature f_6. Actual MI (top) - Black and White MI (bottom)

distinguish. The information stored by each discriminator do not overlap and do not interfere with other discriminators information. This WiSARD peculiarity is extremely important for developments we are introducing in this paper. DRASiW [10, 11] is an extension of the WiSARD model with the capability of storing the frequencies of observed patterns during the training phase in an internal data structure called "Mental Image". The MI can be seen as a grayscale pictorial representation of learned knowledge (implicit knowledge) about a particular class.

Exploiting the content of MI*s* and mainly the meaning associated to them, we propose a procedure to extract a sort of DRASiW "common sense" rules. These rules allow the system to automatically select (activate/deactivate) the discriminators that have to take part to the classification phase.

3 From "Mental Images" to rules

DRASiW MI*s* represent the implicit knowledge that has been acquired about a particular domain after the training phase. From this knowledge, some general rules can be extracted to facilitate the manipulation and use of this "expertise": shortcuts on reasoning, acquired beliefs, quicker and better classification, and so on. Therefore, these rules can be considered as the "common sense" rules of DRASiW about that domain.

Let consider the *Otdigits* dataset, which consists of 10 classes and 64 features with a resolution of 512 tics. After the training phase and for each discriminator a MI is produced by DRASiW. In this case, 10 MI*s* for the 10 discriminators (10 classes). The dimension of each MI is 64 (number of features) by 512 (tics or thermometer resolution). Figure 1 sketches only the part of the MI relating to the feature f_6 for the ten classes. As one can notice, there are white zones both on the left and on the right side of the BW image in which the feature f_6 takes no value for the corresponding class. For instance, before the threshold τ_1 only instances of the classes "0", "5" and "8" are present. Moreover, after the

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

threshold τ_5 instances of classes "0" and "4" are missing.

Before to state a rule associated to the thresholds, a further check has to be done. If the number of instances (N_4) before τ_4 is less than a certain percentage of the total Number of Instances ($\%NI$), the generation of the associated rule is not taken into account. On the contrary, if the number of instances is greater than $\%NI$, DRASiW states the following rule:

```
if ($f_6 < \tau_4 \wedge N_4 > \%NI$) then
    inhibit $D_1$, $D_2$, $D_3$ and $D_7$ for classification
```

and, where possible, the system generates all the possible rules (in this case 6).

Before the classification phase and depending on the feature values of the input, many of the above rules can fire. This process ends up with a list of the following type:

$$[\langle D_0, n_0\rangle, \langle D_1, n_1\rangle, \ldots, \langle D_{10}, n_{10}\rangle] \tag{1}$$

where D_i represents the discriminator and n_i the number of times (at least 1) D_i has been **inhibited**. $n_i = 0$ means that the corresponding discriminator (D_i) has never been inhibited by any rule. Furthermore, the creation of such a list avoids the problem of having conflicts between rules. It represents just a list of possible candidates to be inhibited from the classification process. The rules act only for selecting the discriminators to be inhibited.

Sorted in descending order on n_i, list 1 represents part of the filter DRASiW will use before the classification phase. In fact, the D_i belonging to the first half of the list 1 will be inhibited by DRASIW and not used for classification. This simple heuristic for the selection of the D_i to be discarded in classification has been established with the intention of avoiding the introduction of another hyper-parameter and the obtained results confirm this choice. Moreover, as a side effect, the inhibition of some discriminator makes the system much faster in the classification phase.

4 Experiments

In order to carried out the experiments, we have chosen 55 standard classification datasets available on the KEEL archive.[1] The chosen datasets are mainly those characterized by a prevalence of numerical attributes (features). We ran two different sets of experiments. The aim of the first one is that of comparing the performance of three different models[2] of the classical WiSARD system with rules to the homologous one without rules.

In the second one, we evaluated the performance of the rDAB classifier [12] (with rules) by examining its behaviour when competing with other 13 well known Machine Learning (ML) methods.[3]

[1] https://sci2s.ugr.es/keel

[2] DRASiW, DAB and rDAB.

[3] Decision Tree, k-nearest neighbors, Logistic Regression (LR) and Multinomial LR do not compare neither in table 2 nor in table 3 because never ranked second.

Taking advantage of the available and already partitioned KEEL datasets, experimental results were collected running both a five-fold and a ten-fold cross-validation (from now on 5cv and 10cv). In order to set the optimal possible configuration for each ML method, a selection process was conducted to identify the best hyper-parameters. This process was carried out using the *GridSearchCV* function of the *Scikit-Learn* library, performing an exhaustive search within the parameter grid through a 5cv and a 10cv. The same was done for the DRASiW systems by varying the bit address, the thermometer resolution (tics) and the number of instances $\%NI$.

The measures selected to compare the system performance, with and without the application of the rules, are: the *f1-score*, the *accuracy* and the *gain*. The *gain* is determined by comparing the different WNN system responses running without rules (S) to those running with rules (S_r) and it is defined as $(S_r - S)/\Delta S$, where ΔS is the maximum achievable increase and it is defined as $\Delta S = 1 - S$. A positive *gain* indicates an improvement of the system performance.

5 Results

Before introducing the results, it is worth mentioning that there are datasets for which: 1) the system does not produce any rules (in particular, 8 datasets for the 5cv and 10 for the 10cv); 2) even if the system produces rules their application do not affect the system performance (21 datasets for the 5cv and 23 for the 10cv); 3) for the remaining datasets (26 for the 5cv and 22 for the 10cv) we collected the results reported in tables 1, 2 and 3.

In table 1 the gains (on *f1-score* and on *accuracy*) achieved by the WNN$_r$ systems on the homologous one without rules are reported. The maximum gain in the 5cv has been reached on the *Wine* dataset by all the systems (previous results: *f1-score*=0.9897 and *accuracy*=0.9887; actual results: *f1-score*=0.9949 and *accuracy*=0.9944), while in the 10cv, by rDAB$_r$ on the *Shuttle* dataset.

In the table 2 and in table 3, respectively for the 5cv and for the 10cv, both the performance of rDAB$_r$ and the 2^{nd} best ranked method are reported. We have chosen only those cases in which rDAB$_r$ improved the rDAB performance and ranked 1^{st}. However, in the ranking drawn up on the 55 datasets and both for 5cv and 10cv, rDAB$_r$ ranked 1^{st} followed by *Random Forest* and *Extra Trees*

		DRASiW$_r$		DAB$_r$		rDAB$_r$	
		f1	acc.	f1	acc.	f1	acc.
5cv	Average gain	0.0551	0.0702	0.0593	0.0732	0.0613	0.0685
	Maximum gain	0.5022	0.5070	0.5022	0.5070	0.5022	0.5070
	Minimum gain	0.0007	0.0028	0.0011	0.0014	0.0015	0.0009
10cv	Average gain	0.0418	0.0429	0.0464	0.0437	0.0416	0.0321
	Maximum gain	0.2033	0.2273	0.2033	0.2273	0.2402	0.1599
	Minimum gain	0.0006	0.0007	0.0018	0.0012	0.0007	0.0010

Table 1: Gains with respect to the systems without rules

both for *f1-score* and for *accuracy* (even much better than the previous results obtained in [13]). It is worth noticing that *Random Forest* and *Extra Trees* are both ensemble ML methods.

The improvement in performance obtained by exploiting the implicit knowledge contained in the MI*s* is well expressed by the results reported in the two Gain columns both for 5cv and 10cv. Furthermore, we would like to underline that with the proposed approach, systems with rules always perform better than or equal to those without rules.

5cv	2^{nd} best ranked method			rDAB$_r$		Gain	
Dataset	Method	f1	acc.	f1	acc.	f1	acc.
australian	Quadratic DA	0.7463	0.7627	0.8733	0.8739	0.5008	0.4687
hepatitis	Random Forest	0.7208	0.8969	0.8207	0.9165	0.3576	0.1908
ionosphere	Random Forest	0.9334	0.9396	0.9547	0.9573	0.3186	0.2939
german	Quadratic DA	0.5894	0.6588	0.7020	0.7730	0.2743	0.3348
automobile	Gradient Boost	0.6920	0.8021	0.7735	0.8072	0.2647	0.0259
wisconsin	Extra Tree	0.9740	0.9761	0.9808	0.9825	0.2621	0.2652
wine	Random Forest	0.9936	0.9931	0.9949	0.9944	0.2000	0.2000
appendicitis	Gussian NB	0.7976	0.8870	0.8356	0.9147	0.1879	0.2452
spectfheart	Linear DA	0.7253	0.8039	0.7598	0.8500	0.1255	0.2353
bands	Extra Tree	0.7376	0.7755	0.7620	0.7865	0.0933	0.0490
winequality_white	Random Forest	0.4198	0.6813	0.4527	0.6946	0.0567	0.0416
winequality_red	Random Forest	0.3825	0.6938	0.3997	0.6973	0.0279	0.0116
ecoli	SVC	0.7385	0.8453	0.7429	0.8572	0.0169	0.0766
marketing	SVC	0.2711	0.3464	0.2768	0.3475	0.0078	0.0016
					Average	0.1924	0.1743

Table 2: 5cv – rDAB$_r$ *vs* other ML methods

10cv	2^{nd} best ranked method			rDAB$_r$		Gain	
Dataset	Method	f1	acc.	f1	acc.	f1	acc.
crx	Random Forest	0.8181	0.8234	0.8671	0.8676	0.2690	0.2505
saheart	Gaussian NB	0.6706	0.6971	0.7486	0.8350	0.2554	0.4395
german	Quadratic DA	0.6034	0.6750	0.7006	0.7780	0.2451	0.3169
movement_libras	MLP	0.8800	0.8861	0.9000	0.9028	0.1666	0.1463
spectfheart	SVC	0.7103	0.8168	0.7486	0.8350	0.1323	0.0995
bands	ExtraTree	0.7163	0.7621	0.7277	0.7629	0.1016	0.0639
ecoli	SVC	0.7642	0.8426	0.7769	0.8602	0.0537	0.1121
winequality_red	ExtraTree	0.3836	0.7129	0.4017	0.7167	0.0293	0.0132
tae	ExtraTree	0.6758	0.6833	0.6853	0.6900	0.0292	0.0211
automobile	Gradient Boost	0.7360	0.7897	0.7372	0.7808	0.0046	0.0541
bupa	Ada Boost	0.7351	0.7497	0.7361	0.7361	0.0037	0.0420
					Average	0.1173	0.1243

Table 3: 10cv – rDAB$_r$ *vs* other ML methods

6 Conclusion

A novel extension of the DRASiW system in which rules are automatically explicited and extracted by its MI*s* has been introduced. These rules give the system the opportunity of inhibiting one or more discriminators right before the classification phase. So doing, one can notice that system performance have effectively improved with respect to the corresponding systems without rules. Furthermore, the comparison of the $r\text{DAB}_r$ performance to those expressed by the other 13 chosen ML methods makes this improvement very noticeable.

References

[1] Yu Zhang, Peter Tiňo, Aleš Leonardis, and Ke Tang. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 5(5):726–742, 2021.

[2] Feng-Lei Fan, Jinjun Xiong, Mengzhou Li, and Ge Wang. On interpretability of artificial neural networks: A survey. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 5(6):741–760, 2021.

[3] Wojciech Samek, Gregoire Montavon, Sebastian Lapuschkin, Christopher J. Anders, and Klaus-Robert Muller. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278, 2021.

[4] Tameru Hailesilassie. Rule extraction algorithm for deep neural networks: A review. *International Journal of Computer Science and Information Security*, 14(7):376–381, 2016.

[5] A.N. Averkin and S.A. Yarushev. Review of research in the field of developing methods to extract rules from artificial neural networks. *Journal of Computer and Systems Sciences International*, 60:966–980, 2021.

[6] B.P.A. Grieco, P.M.V. Lima, M. De Gregorio, and F.M.G. França. Extracting fuzzy rules from "mental" images generated by a modified WISARD perceptron. In *ESANN*, 2009.

[7] P. Coutinho, H.C.C. Carneiro, D.S. Carvalho, and F.M.G. França. Extracting rules from DRASiW's "mental images". In *ESANN*, 2014.

[8] I. Aleksander, W.V. Thomas, and P.A. Bowden. WiSARD a radical step forward in image recognition. *Sensor Review*, 4:120–124, 1984.

[9] I. Aleksander, M. De Gregorio, F.M.G. França, P.M.V. Lima, and H. Morton. A brief introduction to Weightless Neural Systems. In *ESANN*, pages 299–305, 2009.

[10] Massimo De Gregorio. On the reversibility of multi-discriminator systems, Technical Report 125/97, Istituto di Cibernetica-CNR, 1997.

[11] C.M. Soares, C.L.F. da Silva, M. De Gregorio, and F.M.G. Franca. Uma implementação em software do classificador WiSARD. In *V Simpósio Brasileiro de Redes Neurais*, volume 2, pages 225–229, 1998.

[12] Gianluca Coda, Massimo De Gregorio, Antonio Sorgente, and Paolo Vanacore. Improving the DRASiW performance by exploiting its own "mental images". In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 363–368, 2023.

[13] Massimo De Gregorio and Maurizio Giordano. An experimental evaluation of weightless neural networks for multi-class classification. *Applied Soft Computing*, 72:338–354, 2018.