Judges System for Classifiers Specialization

Abdel Rodríguez, Isis Bonet, Ricardo Grau, María M. García

Abstract**—**In this paper we designed and implemented a new ensemble of classifiers based on a sequence of classifiers which were specialized in regions of the training dataset where errors of its trained homologous are concentrated. In order to separate this regions, and to determine the aptitude of each classifier to properly respond to a new case, it was used another set of classifiers built hierarchically. We explored a selection based variant to combine the base classifiers. We validated this model with different base classifiers using 37 training datasets. It was carried out a statistical comparison of these models with the well known Bagging and Boosting, obtaining significantly superior results with the hierarchical ensemble using Multilayer Perceptron as base classifier. Therefore, we demonstrated the efficacy of the proposed ensemble, as well as its applicability to general problems.

Keywords**—**classifiers, delegation, ensemble

I. Introduction

EVERY new emerging classification problem on Bioinformatics applications need to be more accurate. In addition, every day becomes more difficult to achieve this with the well-known simple models. There are techniques arising based on combining multiple models into a single one to achieve better results in those problems that needs more accuracy in classification. These models are known as ensemble of classifiers. In Statistics, and mainly the Artificial Intelligence, it has been advanced enough in this way. There are several models on the literature, some ones that base their operation in diversifying the training dataset for each classifier [1-3], some others managing the parameters of the models [4, 5] or simply combining different models [6] to avoid errors correlation.

In this work a new multiclassifier is presented inspired by the necessity of specializing classifiers in different sectors of the training dataset to achieve an increase of its effectiveness. The problem resides in how to separate these sectors, what is a difficult task keeping in mind the great quantity of features that characterizes most of the problems, and the little information we had of them. For these reason, it becomes necessary to build these regions based on what we really know, the performance of classifiers on each training case.

II. Datasets

In order to validate this model, 11 bases were chosen from the UCI Repository [7] representing Bioinformatics or Biomedical problems. These datasets are shown on Table I.

TABLE I
VALIDATION DATASETS (BIOINFORMATICS OR BIOMEDICAL PROBLEMS)

Dataset	Nominal features	Numeric features	Class labels	Cases
audiology	69	0	24	226
breast-cancer	0	4	3	625
Diabetes	0	8	2	768
Heart-c	7	6	2	303
Heart-h	7	6	2	294
Heart-statlog	0	13	2	270
Horse-colic	4	23	2	300
Hypothyroid	22	7	4	3772
Lung-cancer	56	0	3	32
promoters	57	0	2	106
Yeast	0	8	10	1484

This is a fewer population for any statistical comparison. More bases are needed to give the tests the potential to decide if any significant difference exists between the classical and the novel model. For this reason, 26 general bases were picked up too, so a very varied group was collected. This quantity of bases constitutes an enough piece of data to carry out a correct

TABLE II
GENERAL DATASETS FOR VALIDATION

Dataset	Nominal features	Numeric features	Class labels	Cases
autos	8	17	6	205
balance-scale	4	0	3	625
balloons	4	0	2	76
cars	6	0	4	1728
colic	15	7	2	368
credit-a	9	6	2	690
credit-g	13	7	2	1000
glass	0	9	6	214
hayes-roth	4	0	3	132
hepatitis	13	6	2	155
ionosphere	0	34	2	351
Iris	0	4	3	150
kr-vs-kp	36	0	2	3196
labor	8	8	2	57
lenses	4	0	3	23
liver-disorders	0	6	2	345
lymph	2	16	4	148
monks	6	0	2	415
postoperative	7	1	3	90
segment	0	19	7	2310
shuttle-land	5	1	2	15
sick	12	7	2	3772
sonar	0	60	2	208
soybean	35	0	19	683
vote	16	0	2	435
wine	0	13	3	178

Center of Studies on Informatics, Central University of Las Villas, Santa Clara, Cuba. (phone: 53-42-281515; e-mail: abdelr@uclv.edu.cu, isisb@uclv.edu.cu, rgrau@uclv.edu.cu, mmgarcia@uclv.edu.cu.

statistical analysis of the results that there will be shown. We will test over the whole set, and look for a similar behavior on the Bioinformatics and Biomedical bases. Table II shows the additional data.

III. Ensemble Model

As it was said previously, we are presenting a model built by using training subsets. Therefore, each classifier can be specialized in the region of the base where all previous ones failed. Some techniques already use this method [2, 8]; they reinforce the learning of the classifiers in the cases that they have misclassified. Once carried out the above-mentioned, the challenge consists in how to be able to detect how much sure can we be each classifier can predict the objective feature of a new case correctly, and not to make a simple voting (like previous models) where it is rejected in what region of the training database it is expert each classifier. In order to explain our proposal, we should specify how to obtain the diversity of the base classifiers, avoiding correlation among their errors; and then, we should define the method used to combine their outputs.

A. Diversity

The general idea of the algorithm we are proposing is based on going building a group of classifiers specialized in regions of the training dataset. Therefore, they will leave separating progressively the subsets of cases that have not been still well classified, or in fact, they are not probably well classified, for any classifier in the system. So that, at each step, get focus o the learning interest on the cases that it interests us to learn.

We need to divide the dataset in such a way that each classifier is trained to reach the highest accuracy taking into account its capabilities. Another external classifier will evaluate these capabilities or judge that suggests which classifier can be the more suitable or specific on each subset, or analysis level of the original dataset.

We ensure the diversity by changing the dataset for each classifier, so we decided to use an only one model for the base classifiers, that is, the same classifier is used iteratively in the learning of more and more restricted groups. So, when we speak of the first (second…) classifier, really speaks of the classifier it was trained in the first (second…) training subset. For a specific problem, it demands of course, that we should decide which classifier model will be used, keeping in mind the characteristics of the problem. We should also define the pattern of the simple classifier (external or judge) to evaluate the performance of the base classifiers on each iteration, that is, to separate the cases in the successive subsets of training.

The proposed multiclassifier system is constituted by levels. In the first level, we meet the most general classifier; it is able to classify any type of cases, but at the same time with fewer yields. The following one is limited to the cases that the first one did not probably classify well, therefore it is less general than the previous one, it is not able to classify any kinds of cases, but in those that were trained, it should obtain better results. This process repeats until arriving at the last level, where we meet the classifier that trains with the most specific group, with highest performance in a much-reduced group of cases: those that have been more difficult for the rest of the classifiers. Therefore in this work, the level *i* will be related with the training of the classifier C_i and of the judge-classifier J_i, that evaluates which cases can be probably well classified by C_i and separates the remaining cases to constitute the base of training of the most specific classifier C_{i+1}.

Figure one shows three levels system, that is, three judges (J_0, J_1 and J_2) and four classifiers (C_0, C_1, C_2 and C_3)

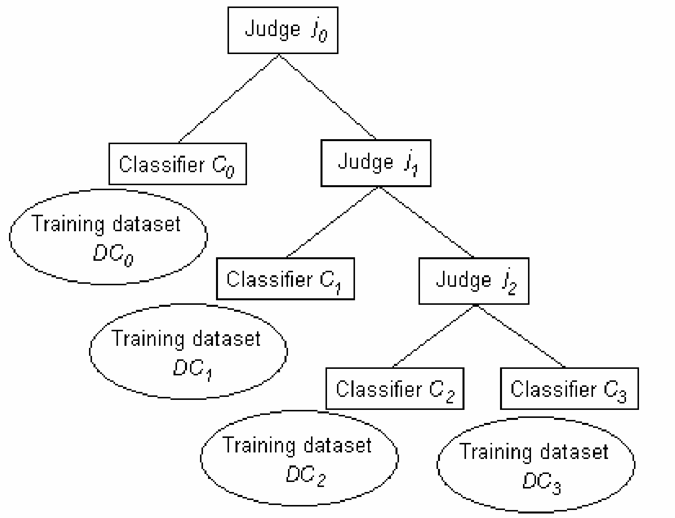

Fig. 1 Three levels ensemble. $DC_0 \subseteq DC_1 \subseteq DC_2 \subseteq DC_3$.

The steps can be formalized as fallows:

1. Train classifier C_i corresponding to level *i*.
2. Train judge-classifier J_i in order to separate those misclassified cases C_i.
3. Construct a new training subset DC_{i+1} with those cases J_i decides will probably be misclassified by C_i.

It is not the same the cases that were misclassified by C_i than the cases that J_i decided as misclassified. If we neglect this difference, the system can be in a fiasco. The ideal conditions, of course, are that J_i decides with 100% of truthfulness, but this is not probable for none of the existing classifiers up to now.

The process should be repeated until completing some stop condition and it can have many to prove. In this work, we will not be plentiful in this condition, we will simply fix the maximum number of levels and we will iterate until arriving to the same one. Therefore, we must add this final step to the procedure described above:

4. If DC_{i+1} is not empty neither the maximum number of iterations is reached, go to the next level having DC_{i+1} as training dataset.

Notice the stop condition can be reached by the last level, or if the classifier of a given level correctly classifies all cases.

B. *Output combination*

Already trained the classifiers, in such a way there is no correlation in their errors, we just need to define how to combine their outputs to classify a new case. The classifiers J_i have been used to define the limits of the training regions for each classifier C_{i+1}, therefore they are those who will take the leading roll in the process of combination of the outputs.

We should have present that is necessary to carry out the combination on the base of the contained knowledge at classifiers J_i. In figure 1 depicts how J_0 separates the training base in two subsets; let us call to whole training base U, and B to the group formed to train C_1. Then the cases that should be classified by C_0 are those that belong to the subset $U \backslash B$, and another specialized classifier must classify B subset: C_1. The same thing happens at each level. We can obtain for each J_i how probable is that the classifier C_i correctly classify a given case or not. Following this idea, the probability P_i that the case x will be well classified by the classifier C_i can be represented as:

$$P_i(x) = (1 - PJ_0(x)) \cdot (1 - PJ_1(x)) \cdots (1 - PJ_{i-1}(x)) \cdot PJ_i(x) \quad (1)$$

Where PJ_i(x) is the probability that the classifier J_i decides that C_i should classify the case x, for $i > 0$. This expression represents the probability that C_i contributes a good answer, conditioned to all the previous ones have possibly responded in wrong way. Clearly P_0(x) = PJ_0(x). Once calculated the conditional probabilities, we will have the system chooses the answer given by the classifier that will respond to each case with more certainty. The learning algorithms and classification are as they continue:

Algorithm 1. Training of the model:

```
1. Initialize $i \leftarrow 0$.
2. Build $C_i$ instance $C$ and train it with $U$.
3. Build $B_i$ from $U$ changing the goal feature as fallows:
    a. Right if $C_i$ well classified the case, or
    b. Wrong in other case.
4. Build $S_i$ instance of $S$ and train it with $B_i$.
5. Suppress from $U$ those cases that $S_i$ classifies as Wrong.
6. If $U$ is empty, $N \leftarrow i$.
7. If $N > i + 1$, $i \leftarrow i + 1$ go to 2.
8. if $U$ is not empty, build $C_{N+1}$ instance of $C$ and train it with $U$
```

Algorithm 2. Classification process by mean of selection:

```
1. For each level $i$, compute by expression (1), the probability $P_i(x)$ that $x$ must be classified by $Ci$
2. Look for the level with the highest value of $P_i$, then classify x by $C_i$.
```

IV. Results and Discusion

In order to validate this method, we present a comparison between the accuracy obtained by our method and results achieved by Bagging and AdaBoost.M1 that are described as very effective and efficient methods in bases of general classification problems in the literature (Opitz and Maclin, 2000). The results are obtained by means of a 10-fold cross-validation. As it is evident, we cannot demonstrate the new model's superiority making a simple comparison of these results. It was used for the same one, the statistical tests of related populations' comparison, in particular non-parametric tests: Friedman (more than two populations) and Wilcoxon (two populations).

TABLE III
Classification accuracy results for Bagging and AdaBoost.M1 models

Datasets	Bagging			AdaBoost.M1		
	J48	SMO	MLP	J48	SMO	MLP
audiology	0.801	0.783	0.819	0.841	0.796	0.845
breast-cancer	0.814	0.872	0.938	0.770	0.869	0.925
diabetes	0.751	0.775	0.763	0.717	0.766	0.759
heart-c	0.776	0.848	0.825	0.815	0.828	0.799
heart-h	0.793	0.844	0.806	0.782	0.847	0.793
heart-statlog	0.804	0.837	0.837	0.785	0.840	0.796
horse-colic	0.710	0.673	0.707	0.717	0.697	0.710
hypothyroid	0.995	0.936	0.952	0.996	0.947	0.950
lung-cancer	0.531	0.438	0.438	0.531	0.344	0.375
promoters	0.840	0.915	0.906	0.915	0.943	0.906
yeast	0.590	0.571	0.596	0.563	0.571	0.577

Tables III and IV sample the results obtained for the accuracy in the classification on Bioinformatics and Biomedical bases, using the models mentioned with a decision tree (J48), Support Vector Machine (SMO) and Multilayer Perceptron (MLP). The statistical validation shows that the results of the three models, using J48 and SMO are, at least comparable.

TABLE IV
Classification Accuracy Results for the proposed model

Datasets	Novel model		
	J48	SMO	MLP
audiology	0.721	0.819	0.845
breast-cancer	0.959	0.969	0.963
diabetes	0.764	0.775	0.764
heart-c	0.785	0.835	0.828
heart-h	0.813	0.857	0.820
heart-statlog	0.807	0.841	0.815
horse-colic	0.683	0.707	0.743
hypothyroid	0.995	0.936	0.953
lung-cancer	0.777	0.892	0.858
promoters	0.858	0.934	0.943
yeast	0.570	0.524	0.594

TABLE V
CLASSIFICATION ACCURACY RESULTS FOR BAGGING AND ADABOOST.M1 MODELS IN THE REST OF DATASETS

Datasets	Bagging			AdaBoost.M1		
	J48	SMO	MLP	J48	SMO	MLP
Autos	0.800	0.712	0.751	0.873	0.737	0.761
balance-scale	0.814	0.872	0.938	0.770	0.869	0.925
balloons	0.724	0.763	0.763	0.776	0.737	0.724
Cars	0.933	0.936	0.997	0.955	0.942	0.994
Colic	0.856	0.832	0.834	0.823	0.804	0.799
Credit-a	0.859	0.848	0.864	0.826	0.826	0.833
Credit-g	0.740	0.748	0.754	0.691	0.747	0.706
Glass	0.734	0.542	0.687	0.776	0.598	0.701
Hayes-roth	0.712	0.803	0.795	0.720	0.818	0.826
hepatitis	0.813	0.858	0.865	0.852	0.832	0.781
ionosphere	0.912	0.877	0.912	0.934	0.889	0.912
Iris	0.953	0.960	0.947	0.953	0.980	0.940
kr-vs-kp	0.993	0.960	0.995	0.996	0.973	0.992
Labor	0.807	0.895	0.912	0.877	0.930	0.912
lenses	0.783	0.696	0.696	0.652	0.826	0.696
liver-disorders	0.722	0.591	0.728	0.661	0.655	0.687
lymph	0.777	0.858	0.838	0.784	0.858	0.838
monks	0.590	0.636	0.547	0.561	0.627	0.564
postoperative	0.689	0.700	0.622	0.578	0.656	0.656
segment	0.976	0.928	0.970	0.982	0.928	0.966
shuttle-land	0.600	0.533	0.600	0.733	0.667	0.667
Sick	0.988	0.939	0.976	0.990	0.952	0.968
Sonar	0.774	0.774	0.841	0.793	0.793	0.846
soybean	0.925	0.931	0.934	0.921	0.922	0.939
Vote	0.970	0.959	0.954	0.954	0.952	0.954
wine	0.933	0.989	0.978	0.949	0.978	0.983

TABLE VI
CLASSIFICATION ACCURACY RESULTS FOR THE PROPOSED MODEL IN THE REST OF DATASETS

Datasets	Novel model		
	J48	SMO	MLP
autos	0.829	0.737	0.771
balance-scale	0.803	0.878	0.920
balloons	0.697	0.763	0.789
cars	0.918	0.933	0.994
colic	0.856	0.845	0.845
credit-a	0.858	0.855	0.843
credit-g	0.717	0.750	0.728
glass	0.692	0.561	0.701
hayes-roth	0.705	0.826	0.818
hepatitis	0.819	0.871	0.832
ionosphere	0.900	0.889	0.920
Iris	0.960	0.967	0.973
kr-vs-kp	0.995	0.962	0.995
labor	0.842	0.982	0.930
lenses	0.848	0.739	0.783
liver-disorders	0.672	0.583	0.707
lymph	0.777	0.892	0.858
monks	0.614	0.627	0.576
postoperative	0.711	0.689	0.589
segment	0.970	0.923	0.968
shuttle-land	0.600	0.600	0.689
sick	0.989	0.939	0.973
sonar	0.764	0.779	0.841
soybean	0.922	0.934	0.939
vote	0.970	0.959	0.959
wine	0.923	0.987	0.983

Tables V and VI show the other results. Statistical tests showed that significant difference exists in favor of the proposed ensemble using MLP when being also compared with their homologous with MLP. Finally, this last one was compared with the previous models, using the three classification models, as base classifiers, and the differences were significant.

V. CONCLUSIONS

In this paper, we presented a new ensemble of classifiers applicable to general problems, and in particular to Bioinformatics and Biomedical problems. It was built based on the specialization of the base classifiers on regions on the training Dataset divided by judges taking into account the local performance of the classifiers. Finally, it was statistically proved the novelty model is useful for different sort of problems, offering better results than Bagging and AdaBoost.M1.

ACKNOWLEDGMENT

This work was developed in the framework of a collaboration program supported by VLIR (Vlaamse InterUniversitaire Raad, Flemish Interuniversity Council, Belgium).

REFERENCES

[1] L. Breiman, Bagging predictors. Machine Learning, 1996. 24: p. 123-140.

[2] Y. Freund and R. E. Schapire, Experiments with a new boosting algorithm. Thirteenth International Conference on Machine Learning, 1996: p. 148-156.

[3] R. E. Schapire, The strength of weak learnability. Machine Learning, 1990. 5(2): p. 197-227.

[4] R. A. Jacobs, S. J. Nowlan, and G. E. Hinton, Adaptative mixtures of local experts. Neural Computation, 1991. 3: p. 79-87.

[5] M. J. Jordan and R. A. Jacobs, Hirarchical mixtures of experts and the EM algorithm. Neural Computation, 1994. 6: p. 79-87.

[6] D. Wolpert, Stacked generalization. Neural Networks, 1992. 5(2): p. 241-259.

[7] D. J. N. A. Asuncion. UCI Machine Learning Repository. 2007.

[8] C. Ferri, P. Flach, and J. Hernández-Orallo. Delagating Classifiers. in 21st International Conference on Machine Learning. 2004. Canada.