A Multi-agent Ensemble of Classifiers

Jaime Calderón, Omar López-Ortega(✉), and Félix Agustín Castro-Espinoza

Área Académica de Sistemas Computacionales,
Universidad Autónoma del Estado de Hidalgo, Carretera Pachuca-Tulancingo,
km. 4.5, C. U., 42084 Pachuca, Hidalgo, Mexico
jaime@hotmail.es, lopezo@uaeh.edu.mx, fcastroe@gmail.com

Abstract. It is well-known that every classifier method or algorithm, being Multi-Layer Perceptrons, Decisions Trees or the like, are heavily dependent on data. That is to say, their performance varies significantly whether training data is balanced or not, multi-class or binary, or if classes are defined by numeric or symbolic variables. Some unwanted issues arise, for example, classifiers might be over-trained, or they could present bias or variance, all of which lead to poor performance. The classifiers performance can be analyzed by metrics such as specificity, sensitivity, F-Measure, or the area under the ROC curve. Ensembles of Classifiers are proposed as a means to improve classifications tasks. Classical approaches include Boosting, Bagging and Stacking. However, they do not present cooperation among the base classifiers to achieve a superior global performance. For example, it is desirable that individual classifiers are able to communicate each other what tuples are classified correctly and which are not so errors are not duplicated. We propose an Ensemble of Classifiers that relies on a cooperation mechanism to iteratively improve the performance of both, base classifiers and ensemble. Information Fusion is used to reach a decision. The ensemble is implemented as a Multi-Agent System (MAS), programmed on the JADE platform. The base classifiers are taken from WEKA, as well as the calculation of the performance metrics. We prove the ensemble with a real dataset that is unbalanced, multi-class, and high-dimensional, obtained from a psychoacoustics study.

Keywords: Classifiers ensemble · Multi-agent systems · Information fusion · Cooperation

1 Introduction

One of the most common data mining tasks consists in assigning a set of inputs to a given class or classes, for which it is required a statistical model representing a mapping from input data (normally described by several attributes) to the appropriate category. This model approximates the true mapping from inputs to outputs. A decision of to which category a new, unseen input belongs, can be reached [11].

G. Sidorov and S.N. Galicia-Haro (Eds.): MICAI 2015, Part I, LNAI 9413, pp. 499–508, 2015.
DOI: 10.1007/978-3-319-27060-9_41

Classification focuses on methods that establish a dependency within data, concentrating on the so-called target attribute [1]. Classification algorithms are intended for modeling the dependency between input attributes and the target attribute. Each object $\mathbf{x}$ is described by its attributes, which in turn define the value of the target attribute $\mathbf{Y}$. Thus, the dataset D from which the model is constructed consists of a finite set of tuples such as $D = \{(x_i, Y_i),\, i = 1 \cdots n\}$. If the target attribute Y possesses nominal values, we are dealing with a classification problem. If the target attribute is described by continuous numeric values, we face a regression problem. Along this text we employ the term *classification* referring to both tasks, and *classifier* to the algorithm that computes *classification*.

However, the usage of a single classifier might not be the best decision to complete a classification task because its performance is affected by several factors i.e. the initial parameters of the algorithm, the distribution of the training dataset, risk of overtraining, among others. This provokes that the arrival of new objects that do not match the statistical model decreases the performance of the classifier. These problems are aggravated when classifiers learn from real datasets, where the distribution of input objects might be unbalanced.

The performance of classifiers is quantified by using the confusion matrix and derived metrics such as the F-Measure and the area under the ROC curve. The F-Measure is used in multi-class problems, while the area under the ROC curve serves only for binary problems. The Weka platform implements the method proposed by Mann Whitney [15] to compute the area of the ROC curve.

It is thought that Ensembles of Classifiers (EoC) have better performance than single classifiers because they benefit from diversity. In an EoC the final conclusion is obtained by aggregating individual decisions. Information fusion is largely employed to that end. As reported in [11] the concept of EoC's has been studied at large: Stacking [4], bagging [2], boosting [5], model averaging and forecast combining are classic methods to form ensembles. A survey of Ensembles of Classifiers can be found in [17].

The main challenges regarding EoC's consist in finding a procedure to employ base classifiers and rules to combine their individual solutions. One of the essential requirements to form ensembles is that two base classifiers do not make the same mistakes on new input data. That is to say, the errors made by the individual predictors must be uncorrelated. For example, if the final solution is obtained by simple majority voting, and if it is assumed that the mistakes made by the classifiers are independent, then the ensemble will misclassify a new input object only if more than half the base classifiers make mistakes. This situation in highly unlikely in heterogeneous EoC.

Thus, the design of EoC's should include a set of base classifiers so the ensemble yields the highest possible performance [13]. To increase the efficiency of the EoC it has been suggested that each base classifier learn from a subset of the original dataset, without duplicating training subsets [7].

We explore the multi-agent paradigm to form ensembles of classifiers. We call this approach a Multi-Agent Ensemble of Classifiers (MAEoC). A software agent is a computer system that is situated in some environment and possesses a strong notion of agency (self-directed behavior). That is to say, an agent is capable of performing autonomous actions within an environment [14,16]. A Multi-Agent System (MAS) consists of agents that respond to changes in the environment and interact with other agents by using a communication language, such as the Agent communication Language (ACL).

MAS are suitable for building dynamic ensembles of classifiers: It is feasible to develop an environment with a number of *classifier agents* where each of them performs its duties (a classification task), communicate to other agents its results (what instances where correctly and incorrectly classified), and learn from what other *classifier agents* do well. Classifier agents complement each other. To the best of the authors' knowledge, MAEoC constitute a novel approach to designing Ensembles of Classifiers.

In Sect. 2 we describe the main notions to form EoC with MAS. We test our MASEoC on a demanding classification task: To determine to which emotional tag some well-defined input parameters used to create fractal music belong. The dataset for this task is described in Sect. 3. The experimental results and comparisons with other types of ensembles is given is Sect. 4. We apply our MAEoC to assist in the creation of musical fragments. Conclusions and a roadmap leading to improvements are delineated in Sect. 5.

2 The Multi-agent Ensemble of Classifiers

The algorithm we propose to form an Ensemble of Classifiers relies on two premises: (i) the performance of base classifiers and (ii) the communication of hits (H) and failures (F) obtained by base classifiers. The steps of the algorithm are outlined next:

1. At time t = 0
   - Coordinator agent recruits m classifiers, $m > 2$, and launches m classifier agents.
   - Coordinator agent broadcasts dataset D to classifier agent$_i$, $\forall i, i \cdots m$.
   - Classifier agent$_i$ performs a ten fold cross-validation. F-Measure$_i$ is calculated.
   - Classifier agent$_i$, $\forall i, i \cdots m$, informs Coordinator Agent two subsets. Subset H_i contains objects correctly classified; subset F_i contains objects incorrectly classified.
2. At time t = 1
   - Coordinator agent forms two aggregated sets: AH and AF. $AH = \cup H_i$; $AF = \cup F_i$.
   - Coordinator agent launches classifier$_{m+1}$, based on the highest F-Measure$_i$ obtained at $t = 0$.
   - Classifier$_{m+1}$ is trained with set AF. Model M_{m+1} is added to the ensemble. F-Measure C$_{m+1}$ is obtained by ten fold cross validation on AF.
   - Classifiers$_{1 \cdots m}$ are trained with set AH. Models $M_{1 \cdots m}$ are added to the ensemble. F-Measures$_{1 \cdots m}$ are obtained by ten fold cross-validation on AH.

3. At time t = 2
   – Classifiers$_{1 \cdots m+1}$ are given weights according to their updated F-Measure at $t = 1$.
   – Weighted voting is used to reach a final conclusion.

To validate our EoC, we perform a classification task whose objective is to assign an input object x_{input} to one of the sixteen emotional tags of Russell's Circumplex Model of Affect [12]. Input object x_{input} possesses attributes to create fractal music as described in [9]. Target attribute Y is the emotional tag.

3 The Psychoacoustics Dataset

To test our ensemble we employ a psychoacoustic dataset, which contains the emotional responses to fractal music, as reported in [8]. It is a challenging dataset because:

– It is a multi-class dataset. There are sixteen different emotions as values of the target variable.
– It is an unbalanced dataset.
– It is a high-dimension dataset. Each object x_i is defined by fifteen attributes, on which fractal musical fragments are created.
– Each object x_i is defined by mixed attributes i.e. nominal and numeric values.
– The target attribute Y_i is an emotional tag associated to each object x_i.

Thus, $D = \{(x_i, Y_i),\ i = 1 \cdots n\}$ contains diverse combinations of input parameters, and the corresponding emotional tag. Input parameters refer to the chaotic system i.e. Lorenz equations, and musical parameters.

The Lorenz equations [10] are defined by variables x, y, and z, which represent the initial values of the attractor. In this case such variables represent initial notes. Variables $sigma$, r and b help determine the actual trajectory. Table 1 displays the range of values that were used to compute the Lorenz attractor (as a generator of melodic sequences).

To complete the creation of musical fragments, musical parameters are paired with the sequence yielded b the Lorenz equations. Musical parameters are: Tempos, Notes Durations, Musical Scales, Chords and Instruments.

Values for variable Tempo are in the interval (60, 220) beats per minute. Notes Durations are tied to Tempos, so they are expressed in values such as whole duration (1), half duration (1/2), a quarter (1/4), and so on. For the present study, Notes Durations lie between 1/16 and 1. Instruments are: (i) Grand Piano, Bright Acoustic and Harpsichord; (ii) Acoustic Guitar, Steel String Guitar;

Table 1. Range of Lorenz Parameters

Lorenz Parameters	
x y z	sigma r b
[0,127]	[-200, 200]

Table 2. Sample of the psychoacoustics dataset

Instrument	Scale	Chord	TempoX	TempoY	TempoZ	DuraX	DuraY	DuraZ	x	y	z	sigma	r	b	Emotion
Piano	PGMayor	none	145	145	150	0.1	0.12	0.2	12	15	16	4	23	2	Happiness
Piano	CNMenor	major	140	165	150	0.125	0.1	0.25	23	56	16	4	23	2	Sadness
Distortion Guitar	PGMayor	none	125	135	145	0.125	0.115	0.13	23	34	67	4	8	7	Stress
Guitar Harmonics	PEMenor	none	187	180	199	0.175	0.195	0.145	63	40	77	30	57	11	Excitation
PizzicatoStrings	PCMenor	none	187	180	199	0.205	0.275	0.345	65	41	79	40	23	115	At ease
Overdriven Gtr	GAMenorr	major	255	255	255	1	1	1	80	119	90	43	14	37	Stress
String ensemble	PGMenor	major	255	255	255	0.125	0.13	0.12	67	65	68	100	100	100	Euphoria
Electric Muted Gtr	PEMenor	diminished	255	255	255	0.132	0.137	0.131	12	120	92	200	200	200	Astonishment
StringEnsemble	PGMayor	none	124	124	124	0.55	0.5	0.45	15	15	15	-10	-10	-10	Droopiness
Piano	PEMenor	minor	90	90	90	0.065	0.065	0.065	12	100	45	-3	3	-3	Satisfied
Overdriven Gtr	PGMenor	add9	80	80	80	0.125	0.125	0.125	120	12	24	-65	-65	-65	Tiredness
Piano	BMelMinor	none	60	60	60	.125	.125	.125	23	45	78	-100	-100	-100	At ease
Bright Acoustic	CMelMinor	Minor	60	60	60	.125	.125	.125	67	10	9	-100	-100	-100	At ease
Bright Acoustic	CMelMinor	Minor	60	60	60	.125	.125	.125	67	10	9	-100	-100	-100	Tiredness
Piano	GBlues	Major	220	220	220	0.125	0.125	0.125	61	9	3	-150	-150	-150	Euphoria
Cello	CBlues	Minor	140	140	140	0.125	0.125	0.125	2	2	2	150	150	150	Anger

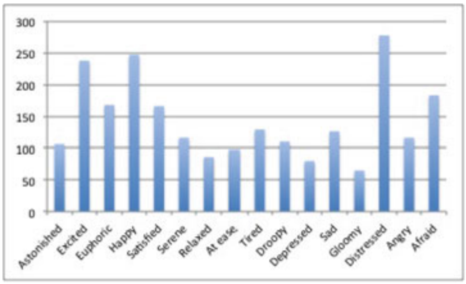

Fig. 1. Distribution of evaluations by emotional label

(iii) Electric Clean Guitar, Electric Jazz Guitar, Guitar Harmonics, Distorted Guitar, Overdriven Guitar, Electric Muted Guitar; (iv) Violin, Viola, Cello, Tremolo, Pizzicato, Orchestral Strings, String Ensembles; and Acoustic Bass, Electric Bass Finger, Electric Bass Pick, Fretless Bass, Slap Basses, Contrabass.

The Chord variable accepts the following values: Mayor chords; Minor chords; Augmented chords; Diminished Chords; Other chords, and No chords. Musical Scales are: Pentatonic Scales; Harmonic Scales; Natural Scales; Blues Scales; Melodic Scales; No scale.

Some $(x_i, Y_i) \in D$ are shown in Table 2. Altogether, dataset D contains 2312 objects at the time of performing the experiments. Figure 1 illustrates the distribution of objects by emotional label.

4 Experimental Results

The proposed algorithm is implemented on the JADE platform [3]. Classifiers and performance measures are obtained from Weka [6]. The psychoacoustics dataset is stored on a relational database implemented in the MySQL database management system. We now present the experimental results. To do so, we follow the steps of the proposed algorithm (see Sect. 2).

The Multi-Agent Ensemble of Classifiers (MAEoC) is composed of the following base classifiers C_i: (i) Naive Bayes, (ii) k-Nearest Neighbors, $k = 5$,

Table 3. Performance of the EoC vs other techniques

C_i	C_{it0}	Bagging	Boosting	Stacking Meta Ci	Oversampling	Undersampling	C_{it1}
NB	0.129	0.13	0.129	0.15	0.146	0.105	0.181
5-NN	0.201	0.201	0.201	0.167	0.305	0.154	0.277
J48	0.161	0.164	0.161	0.151	0.273	0.146	0.236
SVM	0.183	0.184	0.183	0.189	0.243	0.157	0.248
MLP	0.191	0.199	0.191	0.171	0.261	0.156	0.253
MAEoC 0.337							

(iii) Decision Tree J48, (iv) Support Vector Machine, and (v) Multi-Layer Perceptron. This ensemble ensures diversity of algorithms. The main results are summarized in Table 3.

Firstly, the F-Measure was quantified for each classifier C_i when they were trained with original dataset D. These results are presented in column C_{it0}.

A second experiment consisted in forming ensembles using bagging, boosting, and stacking on every classifier C_i.

Since dataset D is unbalanced, we modify it via Oversampling and Undersampling. Then each classifier C_i was trained with those modified datasets. The F-Measures are given in columns Oversampling and Undersampling, respectively.

We also present how the MAEoC behaves at time $t = 1$. Column C_{it1} shows the F-Measure when base classifiers C_i are trained with dataset AH. We urge reader to remember that AH is the aggregated dataset of the objects correctly classified.

Finally, the performance of the MAEoC is given.

The performance of the MAEoC can be observed in Figs. 2 and 3.

We normalized the performances based on the highest F-Measure obtained when the base classifiers were trained with original dataset D. In this experiment, that honor corresponds to k-Nearest Neighbors, $k = 5$. F-Measure = 0.201

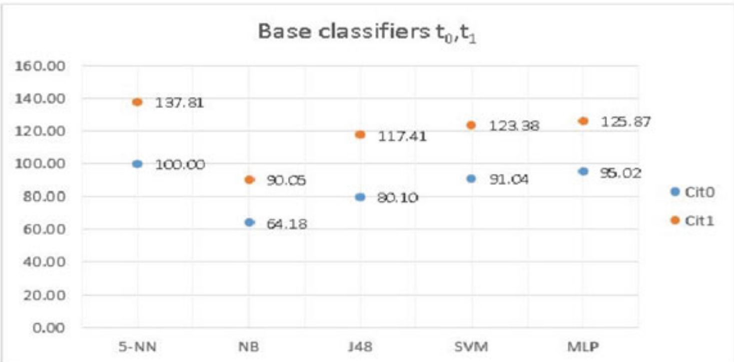

Fig. 2. Performance of classifiers at t = 0 and t = 1 (Color figure online)

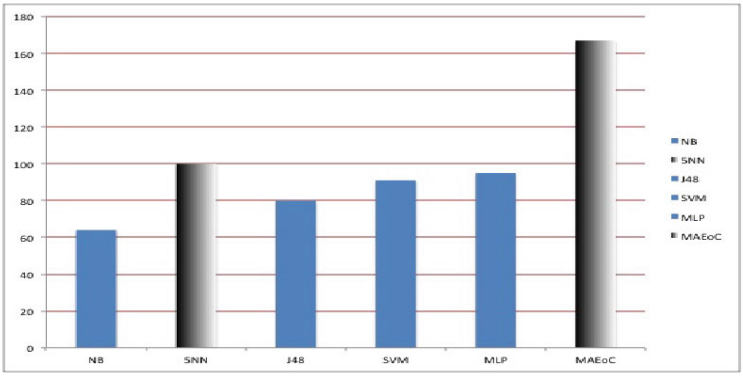

Fig. 3. Performance of base classifiers vs MAEoC

corresponds to a baseline of 100. In Fig. 2 the blue dots correspond to the normalized F-Measure obtained by the base classifiers when they were trained with the original dataset D. Thus, it can be seen that the Naive - Bayes classifier (NB) has a performance 36 % worse than 5-NN. Conversely, The red dots reflect the normalized F-Measure when the base classifiers were trained with dataset AH, that is to say, the aggregated set of correctly classified instances. Based on this data, we support our claim that communicating hits and fails is a way of reducing the influence of the data distribution in the creation of its statistical model.

Figure 3 presents a comparison of the F-Measures obtained at t = 0 and the final F-Measure obtained with the MAEoC, which improved in the order of 67 %.

4.1 Application to Computer-Assisted Creativity

The application to Computer-Assisted Creativity is presented in the following screenshots. As one of the applications of the MAEoC, we use it to help a creative subject to know what emotions will most likely be evoked by the parameters s/he enters in order to create a musical fragment. The MAEoC is used as follows. Once the MAEoC is launched, the training phase begins. This is show in Fig. 4. As stated before, dataset D contains the emotional responses to fractal music obtained in a psychoacoustics study [8]. In the let frame of the screenshot the communication among classifier agents and coordinator agent is shown.

As soon as the MAEoC is trained, the final weights are given to each classifier. This includes the classifier $m + 1$ contemplated as part of the algorithm. These results are show in Fig. 5.

When the voting weights are assigned to the classifiers, the MAEoC is ready to classify new objects. This is illustrated in Fig. 6. Variables x, y, and z are the initial values of the Lorenz attractor. Variables $sigma$, r, and b define its trajec-

Fig. 4. Training of the MAEoC

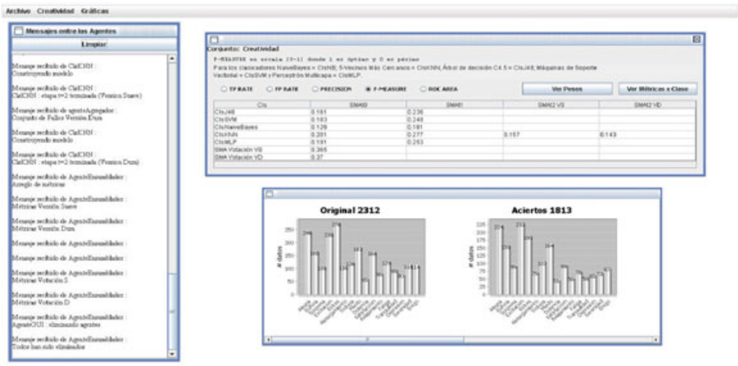

Fig. 5. Voting weights obtained by the members of the ensemble

Fig. 6. Example 1. Classification of a new input object

tory. The remaining of the variables refer to the musical parameters necessary to create a musical fragment: Instrument (Instrumento) is Piano; Chord (acorde) is null; Musical Scales (Escala) is set to be G Major Pentatonic. Variables Tempo and Notes Durations defined the rhythm of the musical fragment. The MAEoC classifies this new input object X_i belongs to the class of objects that evoke *happiness* (alegría).

Fig. 7. Example 2. Classification of a new input object

If that were the creative subject's intention, then those input parameters would be used to render musical fragments; otherwise, the creative subject would be free to change their values. Figure 7 shows a different input object. The MAEoC determines that it belongs to the class of objects that provoke *Distress* (estres).

When the MAEoC is incorporated into a system that creates music, it guides the creative subject in her/his endeavor. The MAEoC classifies newly input values into one of the sixteen classes of the CMoA, preventing the creative subject from doing educated guesses.

5 Conclusions and Future Work

We propose a Multi-Agent Ensemble of Classifiers (MAEoC). The algorithm employed to construct and exploit such ensemble is based on the dynamic calculation of a performance measure, communication, and information fusion. We tested our proposal on a demanding classification tasks: determine the most likely emotional tag based on a number of musical and fractal parameters on which a musical fragment is created. A psychoacoustics dataset is used as training source. We also compared he MAEoC with aggregation techniques (bagging, boosting, stacking), and with data manipulation techniques (undersampling and oversampling). On our experiments the MAEoC obtained the highest F-Measure.

Future work includes development of different versions of the algorithm. For instance, we can proposed a version where objects in subset *AH* are weighted according to the number of classifiers that predicted them correctly. We will test the MAEoC with different datasets, taken mostly from the UCI repository. We will continue acquiring data regarding the emotional responses, and train the MAEoC with upgraded versions of the dataset.

References

1. Berthold, M.R., Borgelt, C., Hoppner, F., Klawonn, F.: Guide to Intelligent Data Analysis, 1st edn. Springer, London (2010)
2. Breiman, L.: Bagging predictors. Mach. Learning **24**, 123–140 (1996)

3. Caire, F.L.B.G., Greenwood, D.: Developing Multi-Agent Systems with JADE. Wiley, New York (2007)
4. Dzeroski, S., Zenko, B.: Is combining classifiers with stacking better than selecting the best one? Mach. Learning **54**, 255–273 (2004)
5. Freund, Y., Saphire, R.: Experiments with a new Boosting algorithm. In: Proceedings of the Thirteen International Conference on Machine Learning, pp. 148–156 (1996)
6. Holmes, G., Donkin, A., Witten, I.: Weka: a machine learning workbench. In: Proceedings of the Second Australia and New Zealand Conference on Intelligent Information Systems, Brisbane, Australia (1994)
7. Kheradpisheh, S.R., Sharifizadeh, F., Nowzari-Dalini, A., Ganjtabesh, M., Ebrahimpour, R.: Mixture of feature specified experts. Inf. Fusion **9**, 4–20 (2014)
8. Lopez-Ortega, O., Franco-Arcega, A.: Analysis of psychoacoustic responses to digital music for enhancing autonomous creative systems. Appl. Acoust. **89**, 320–332 (2015)
9. López-Ortega, O., López-Popa, S.I.: Fractals, fuzzy logic and expert systems to assist in the construction of musical pieces. Expert Syst. Appl. **39**, 11911–11923 (2012)
10. Lorenz, E.N.: Deterministic non-periodic flow. Atmos. Sci. **20**, 130–141 (1963)
11. Oza, N.C., Tumer, K.: Classifier ensembles: select real-world applications. Inf. Fusion **9**, 4–20 (2008)
12. Russell, J.A.: A circumflex model of affect. J. Pers. Soc. Psychol. **39**, 1161–1178 (1980)
13. Ruta, D., Gabrys, B.: Classifier selection for majority voting. Inf. Fusion **6**, 63–81 (2005)
14. Weiss, G.: Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence, 1st edn. The MIT Press, Cambridge (1999)
15. Witten, I., Frank, E., Hall, M.: Data mining. Practical Machine Learning tools and Techniques, 3rd edn. Morgan Kaufmann, San Francisco (2011)
16. Wooldridge, M.: An Introduction to MultiAgent Systems, 2nd edn. Wiley, New York (2009)
17. Wozniak, M., Grana, M., Corchado, E.: A survey of multiple classifier systems as hybrid systems. Inf. Fusion **16**, 3–17 (2014)