Neurocomputing 231 (2017) 3–10

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Virtual organization with fusion knowledge in odor classification

Gabriel Villarrubia[a], Juan F. De Paz[a,*], Dechen Pelki[b], Fernando de la Prieta[a], Sigeru Omatu[c]

[a] Biomedical Research Institute of Salamanca/BISITE Research Group, University of Salamanca, Edificio I+D+I, 37008 Salamanca, Spain
[b] Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Campus Montegancedo, 28660, Boadilla del Monte, Madrid, Spain
[c] Osaka Institute of Technology, 5-16-1 Omiya, Asahi-ku, Osaka 535-8585, Japan

ARTICLE INFO

Keywords:
Virtual organization
Neural networks
Classifiers
E-nose

ABSTRACT

E-nose systems are becoming increasingly important instruments across all industries, especially the fields of food and beverages and biomedicine. Given the inaccurate, unsafe and unreliable dependency on the human nose to detect smells that are highly risky and hazardous to human health, e-nose systems offer a tremendous advantage. E-noses are convenient, highly efficient and can be used in real life to detect various types of odors. This paper presents a virtual organization of agents that integrates different classification techniques and neural networks to perform information fusion from parameters retrieved by the E-nose. The integral brain in e-noses is the data processing system, which classifies odors that have been detected by the detection part of its system. The system mimics how a human brain classifies odors.

1. Introduction

Electronic noses (E-noses) are modern electronic odor classification technologies based on olfactory techniques that detect and classify volatile odor. The odor receptors in a biological nose are replaced by gas sensors, which preprocess and identify the odor using a computer and software [3]. E-noses generally consist of an array of sensors which depend on diverse functionalities, such as the change in the thickness of the film of the semiconductor sensors when exposed to different gases [4]. E-noses also consist of an electronic circuitry, a sampling system, and data analysis software. It is necessary to incorporate new techniques that perform information fusion on the information obtained from several experts [1,5] in a way similar to how human experts detect and classify odors.

E-noses are becoming increasingly popular in various fields such as the food industry, to improve the quality and safety of food processing [6], the biomedical field, to advance the effectiveness and efficiency of biomedical treatments and healthcare services, [7] and many other sectors. Compared to the other techniques used for detecting odor, e-noses uses samples of existing odors to classify the odor, while existing techniques such as gas chromatography and the mass spectrometry, separate the odor with the aroma in its components from the mixture, and then identify each component by comparing them with a standard component [4]. An odor stimulus will generate patterns or fingerprints (or smellprints) from the components to construct a database and train a pattern recognition system so that unknown odors can be classified and identified [8].

This paper presents the use of virtual organizations of agents to process information in a similar way to humans. The system would incorporate roles that would manage the information retrieved from the sensors to create patterns, which has been done in previous works on indoor locating systems [5]. The agents incorporate roles that process the information by detecting patterns. This information is then processed by others agents that fuse the predictions made by agents. The agente incorporates the CBR model in order to classify the different patterns. The retrieve phase use a ESOINN neural network [2] to organize the database in clusters. To implement the reuse phase in the CBR cycle, several classifiers will be included to detect patterns, and neural networks with several configurations will be used to fuse the information provided by the classifiers. The system is applied over a case study to analyze the performance according to various configurations created in the virtual organization.

This paper is organized as follows: Section 2 revises the related work, Section 3 describes the proposed architecture, Section 4 presents the case study in which the platform is applied, and finally Section 5 shows the results and conclusions obtained.

2. State of the art

According to Chen et al. in [9], an E-nose is an instrument used for the automated detection and classification of odors, vapors and gases, thus mimicking the human olfactory apparatus. Gardner and Bartlett in [10] describe an e-nose as an instrument that comprises an array of electronic chemical sensors with partial specificity and an appropriate

* Corresponding author at: University of Salamanca, Computer Science Department, Biomedical Research Institute of Salamanca/BISITE Research Group, 37008 Salamanca, Spain.
E-mail addresses: gvg@usal.es (G. Villarrubia), fcofds@usal.es (J.F. De Paz), pelki92sag@gmail.com (D. Pelki), fer@usal.es (F. de la Prieta), omatu@rsh.oit.ac.jp (S. Omatu).

http://dx.doi.org/10.1016/j.neucom.2016.05.106
Received 29 November 2015; Received in revised form 30 April 2016; Accepted 5 May 2016
Available online 15 October 2016
0925-2312/ © 2016 Elsevier B.V. All rights reserved.

pattern-recognition system, which is capable of recognizing simple or complex odors.

E-noses use qualitative, low-cost, real-time and portable methods to perform reliable, objective and reproducible measures of volatile compounds and odors, and it is important to know the differences between an artificial system and the physiology required to develop such a system [11].

In the work [12] Rodríguez et al. present an e-nose called the A-Nose. The data response from the A-Nose of Colombian coffee is divided into simple and complex odors using PCA; it is then validated using the MLP backpropagation with the leave-one-out cross validation method. They achieved a 92.5% success rate in classifying coffee data into 9 samples using the MLP backpropagation with the LOO cross-validation technique. It is an interactive validation approach that generates *N* evaluations for *N* procedures (1 for each measurement) so that the final result is the average success of the entire iterative process. However, it is difficult when the sample size increases.

In [13], Yu et al. present a portable e-nose Pen2 in the paper “Quality grade identification of green tea using the eigenvalues of PCA based on the E-nose signals” to identify the quality grade of green tea by extracting feature vectors from the response generated by the EN that are analyzed, reduced and optimized by PCA. Only the front five principal components were extracted and used for grading by LDA and BPNN. However, this method is applicable only for tea beverages and not for tea leaf and tea remains.

In [14], Shilbayeh and Iskandarani use a TGS 800 series Smart electronic Nose in the paper “Quality control of coffee using an E-nose system”. Here, the sensed odor is converted into an electrical signal which is conditioned and sent to a computer to be interpreted and classified using a Backpropagation Neural Network. The e-nose is able to automatically detect odor as well as allow any of the 800 series sensors to be interfaced without the need for any hardware modification or adjustment. However, use of the BPNN could lead to over-fitting.

In [15], the E-nose and mass spectrometer-based E-nose (MSE-nose) used by Berna et al. in “electronic nose systems to study shelf life and cultivate the effect on tomato aroma profile” compares the 2 e-noses for detecting changes in the tomato aroma profiles of two different crop. They used PCA to plot the discriminations and reported that the mass spectrometer-based e-nose performs better.

In [16], Ali et al. presents a quartz crystal microbalance based electronic nose in the paper “Detection of bacterial contaminated milk by means of a quartz crystal microbalance based electronic nose” where the Quartz Crystal Microbalance (QCM) sensors were used for the headspace analysis of milk volatiles and later PCA to analyze the sensor array responses.

In [17] “Bacteria classification using Cyranose 320 electronic nose”, Dutta et al. uses a Cyranose 320 electronic nose which classifies 6 different bacteria that cause eye infection. They also employ a number of methods such as PCA, using a combination of clustering algorithms (3D-scatter plot, Fuzzy C Means, SOM) and supervised classifiers (Multilayer Perceptron (MLP), Probabilistic Neural Network (PNN) & Radial basis function network (RBF)) for classifying the data. Combining 3 different non-linear classifiers solves the feature extraction problem with very complex data and enhances the performance of the Cyranose 320 e-nose, but it can be very difficult and complex.

In [18] “Improving the Classification accuracy in electronic Noses Using Multi-dimensional combining (MDC)”, Chen et al. use a Cyranose 320 E-nose and propose a Multi-dimensional combining (MDC) method to combine the classification outputs of individual classifiers for household-fragrances. There are two methods: combining feature extraction methods, and combining dimension reduction methods such as the PCA, Independent Component Analysis (IDA), and Multiple Linear Discriminant (MLD). The combination methods for combining the individual classifiers are arithmetic mean average, geometric mean average and squared mean average. MDC is compared with other traditional pattern recognition methods such as the KNN, LDA and PNN concluding that there is an increase in the overall classification accuracy, which could not be achieved by using a single individual classifier.

Therefore, from the state-of-the-art approaches, it can be concluded that the use of the Principal Component Analysis (PCA) as a dimension reduction method for preprocessing the initial data, and the subsequent application of the Back Propagation Neural Networks (BPNN) for classifying an e-nose system proves to be an efficient approach. However, the work by Chen et al. in [18] indicate that the results of combining the different classifiers prove that the classification results and accuracy are greater than those of individual classifiers. Therefore, the different types of classifier methods are studied in order to propose an ensemble of classifiers with maximum accuracy.

3. Proposed system

In order to process and analyze the information, a virtual organization of agents is applied, as its open nature facilitates the creation of new roles with new behaviors, in this case experts or mixture techniques, to test different configurations. The virtual organization of agents is composed of three layers. Layer 0 is the low level layer which retrieves the information from the sensors. Layer 1 does the first processing of the signal with several filters. Finally, layer 2 incorporates three elements: the organization cluster to cluster the cases; the prediction organization with the experts; and the mixture organization that is responsible for obtaining the prediction of the experts. It then combines them to generate the final result. The upper levels are associated with the services that are provided to the users. Fig. 1 shows the system architecture.

The agents of the expert organizations incorporate a case based reasoning mechanism, which is used to make the predictions and learn from the cases newly incorporated in the memory. The definition of a case *j* follows the expression (1).

$$c_j = (s_1, s_2, \ldots s_n, class) \tag{1}$$

where s_i is the input i in the system and class is the final class of the case. Additionally, the cases were grouped into similar cases through an ESOINN network that allows distributing cases in similar cases. Therefore, the case base is grouped according to similar cases using the ESOINN network, so that the memory contains the cases defined by (1) and the clusters according to (2)

$$G = \cup g_i / g_i \subseteq \cup c_j,\ g_l \cap g_m = \varphi \tag{2}$$

Using this memory, the reasoning cycle is constructed for each agent in the prediction organization as shown in Fig. 2. In the retrieve phase the system retrieves the group with the most similar cases created by ESOINN, retrieving the mesh closest to the new case introduced. If the system includes the trained classifier for the recovered cases, it is recovered for its application in the reuse phase. In the reuse phase the agents builds the classifiers with the retrieved cases. If the classifier already exists, it is retrieved and then used to predict the new case. Memory *M* now contains a set of tuples consisting of cluster g_i and the classifier cl_i associated to cluster g_i according to (3).

$$M = \cup (g_i, cl_i) \tag{3}$$

In the revision phase, the prediction is analyzed and corrected if necessary. In the learning phase the new case is stored; the classifier is rebuilt if a misclassification has been made.

3.1. Mixture

A mixture of experts involves the creation of procedures to merge information from the predictions made by different experts. When there are several experts, the fusion must be applied while bearing in

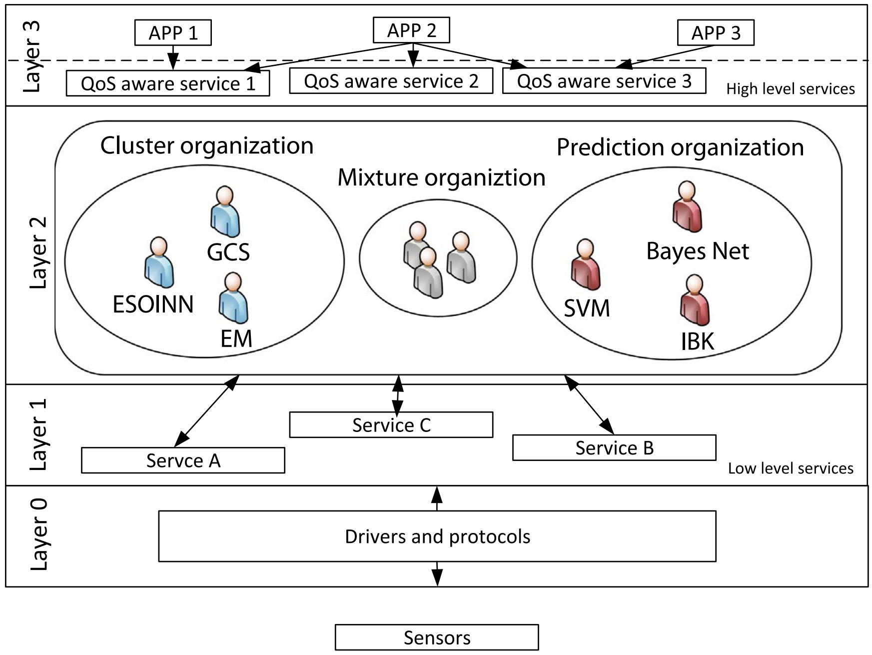


Fig. 1. Virtual Organization of agents.

mind the prediction of each expert. It is necessary to consider not only the number of experts with a certain prediction, but also the reliability of the experts for the analyzed case. Formally, the mixture process involves minimizing the expression (2).

$$\min \sum (y_i - y_i'(cl_i^1(\vec{x}_i), .., cl_i^n(\vec{x}_i)))^2 \tag{4}$$

where y_i is the output value for the input pattern i, y_i' is the prediction value based on the mixture for the pattern i, $\vec{x}_i$ is the input vector I, and $cl_i^k(\vec{x}_i)$ is the ouput value of the classifier k with the input pattern i.

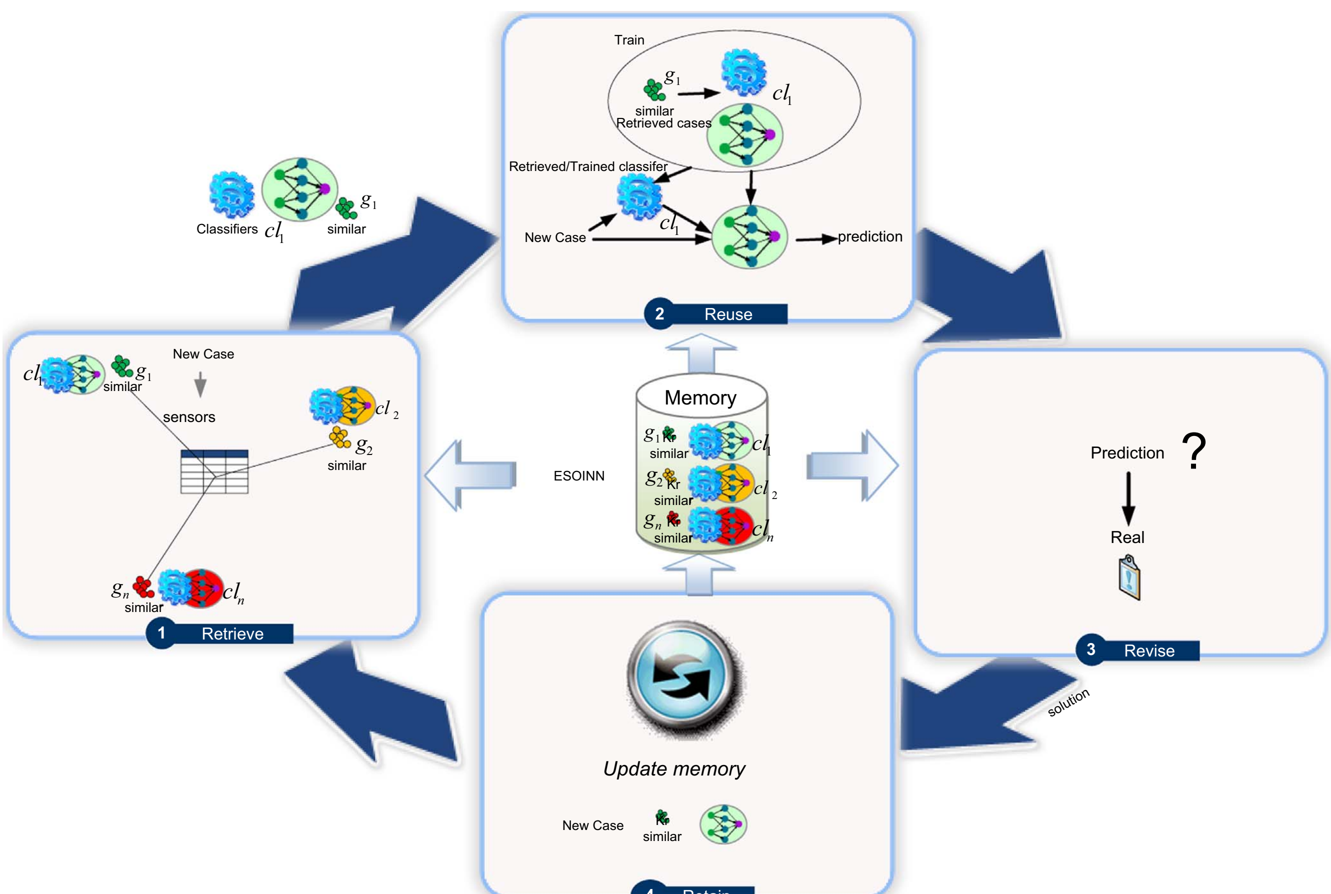


Fig. 2. Case based reasoning followed by the agent in the prediction organization.

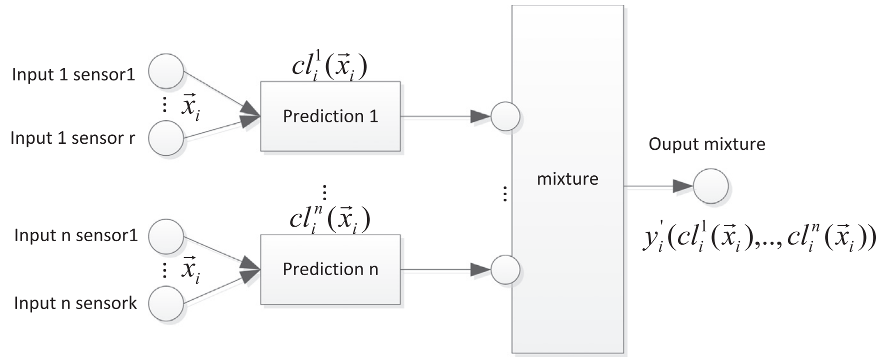


Fig. 3. Mixture of experts without context.

This scheme is represented in Fig. 3. This would be the most common process to fuse the ouput value of the experts.

When working with categorical values, the function can be modified to consider different measures such as the kappa index, the area under roc curve, or the success rate.

$$\max f(\vec{y}, \cup\, y_i'(cl_i^1(\vec{x}_i), ..,cl_i^n(\vec{x}_i))) \tag{5}$$

where f is the function to measure the efficiency, $\vec{y}$ are the classes, and y_i' is the predicted class for pattern i.

The scheme shows deficiencies such as the possibility of i and j patterns for which the vector $(cl_i^1(\vec{x}_i), ..,cl_i^n(\vec{x}_i))$ is equal to $(cl_j^1(\vec{x}_j), ..,cl_j^n(\vec{x}_j))$ but their final classes are different. The mixture of the outputs of the experts cannot distinguish this situation, which makes it necessary to introduce additional information so that the system is able to learn from the errors of each expert. Consequently, the data input by each expert are also found in the mixture, which allows the mixture to be made according to the both the output and the input of the experts. This allows the system to learn which inputs made by each expert are most useful. As a result, we propose maintaining the definition of the system as indicated in (5).

$$\min \sum (y_i - y_i'(\vec{x}_i, cl_i^1(\vec{x}_i), ..,cl_i^n(\vec{x}_i)))^2 \tag{6}$$

Eq. (5) is modified in a similar way to consider the input vector, as shown in Eq. (7)

$$\max f(\vec{y}, \cup\, y_i'(\vec{x}_i, cl_i^1(\vec{x}_i), ..,cl_i^n(\vec{x}_i))) \tag{7}$$

In this case, the graphical representation would be as shown in Fig. 4. In this case, the context is associated with the inputs in the classifiers.

There are a number of different mixtures techniques, making it possible to apply a classifier, linear programming or other options such as neural networks. In this case, an MLP is proposed since it has no restrictions and, furthermore, it allows the optimization of any functions even if they are not linear. In this case, the function is defined according to (4).

$$\min \sum (y_i - MLP_i'(\vec{x}_i, c_i^1(\vec{x}_i), ..,c_i^n(\vec{x}_i)))^2 \tag{8}$$

Likewise, the process of finding the minimum will be made by a backpropagation algorithm.

The configuration of the MLP is performed so that the number of the hidden layers is $2n$+1 where n is the number of sensors input layer plus the number of experts system. The use of bias is incorporated to the network and the activation function is sigmoidal. Moreover, the values in the inputs are redefined within the range [0.2, 0.8] to avoid extreme values in the sigmoidal function during the training.

The function to update weight and bias in the output layers is defined by (9)(10)

$$w_{kj}^p(t+1) = w_{kj}^p(t) + \eta(y_k^p - y_k'^p)(1 - y_k'^p)y_k'^p y_j'^p + \mu(w_{kj}^p(t) - w_{kj}^p(t-1)) \tag{9}$$

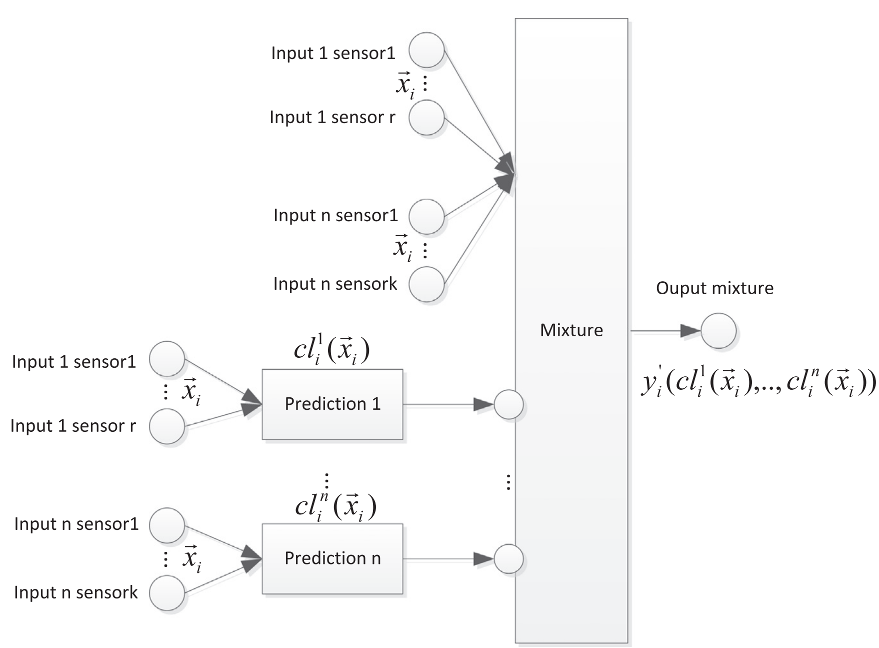


Fig. 4. Mixture of experts considering the input vector.

$$\theta_k^p(t+1) = \theta_k^p(t) + \eta(d_k^p - y_k^p)(1-y_k^p)y_k^p + \mu(\theta_k^p(t) - \theta_k^p(t-1)) \quad (10)$$

where w_{kj}^p is the weight of neuron j in the hidden layer and neuron k in the ouput layer, η is the learning rate, y_k^p the output value in neuron k in the ouput layer, $y_k'^p$ the obtained value in neuron k in the output layer, $y_j'^p$is the value in neuro j in the hidden layer, μ the momentum, t the iteration. θ_k^p is the bias of neuron k in the output layer.

Weights and bias in the hidden layer are updated according to (11) (12)

$$w_{ji}^p(t+1) = w_{ji}^p(t) + \eta\,(1-y_j'^p)y_j'^p \left(\sum_{k=1}^{M} (y_k^p - y_k'^p)(1-y_k'^p)y_k'^p w_{kj}\right) x_i^p + \mu(w_{ji}^p(t) - w_{ji}^p(t-1)) \quad (11)$$

$$\theta_j^p(t+1) = \theta_j^p(t) + \eta(1-y_j'^p)y_j'^p \left(\sum_{k=1}^{M} (y_k^p - y_k'^p)(1-y_k'^p)y_k'^p w_{kj}\right) + \mu(\theta_j^p(t) - \theta_j^p(t-1)) \quad (12)$$

where the subindex i in the variables represents neuron i in the input layer, and subindex j neuron j in the hidden layer, and $x_i^p$the inputs variables.

4. Case study

The e-nose data is collected from the e-nose system developed at OIT in Japan. Five different types of sensors were used to detect four types of odors: Acetaldehyde, Ethylene, Hydrogen Sulphide and Methyl Mercaptan. The experiments were carried out three times to detect different odors.

The data is continuously collected from the sensors, with a sampling period of 0.1 s. For each of the sensors two different values are captured:

– VS: Voltage signal from sensor at the source.
– VRL: Voltage signal from sensor recorded across the load resistor.

Finally, the gas-type is the last attribute of each of the vectors contains the information shown in Table 1. The database contains a total of 25.679 values belonging to 4 different gas-types.

The complete description of the input is defined in Table 2.

All the sensors used for the experiment are of the SB-series as seen in Fig. 5. SB-series models have the following characteristics [47]:

– Compact design
– Low power consumption
– Power supply: 5 V DC
– Output: 0 to 3.5 V DC
– 3 wired connectors, attached (10 cm).

As explained in [19], the SnO_2 semiconductor material is heated to a certain temperature based on the type of gas to detect. When the concentration of the gas changes, the resistance of the sensing material also changes rapidly due to the adsorption/desorption of oxygen and the chemical reactions that take place between the surface oxygen and the gases. Thus, the sensor resistance decreases under the presence of reducing gases such as CO, methane, and hydrogen.

The changing voltages at the source VS and at the resistor load VRL are recorded to detect the types of gases.

Table 1
Format of data used.

CH-1[VS]	CH-2[VS]	CH-3[VS]	CH-4[VS]	CH-5[VS]	CH-1[VRL]	CH-2[VRL]	CH-3[VRL]	CH-4[VRL]	CH-5[VRL]	GAS-TYPE

Table 2
The types of sensors used.

Channels	Sensor Name	Categories	Main detecting gas	Properties
CH-1	SB-15	Gas detector	Propane/ Butane	1. High sensitivity, low sensitivity to noise or gases, quick response speed, strong poisoning resistance and significant low power consumption design (120 mW)
CH-2	SB-EN2	Portable Checkers	Alcohol	1. Only a single gas is monitored
CH-3	SB-EN3	Portable Checkers	Breath	1. Only detects breath
CH-4	SB-42A	Refrigerant	Freon	1. High sensitivity to HFCs (e.g. Freon: R-134a) with improved cross sensitivity to other gases. 2. Suitable for R-134a, R-410a, R-407c and other new Freon families which contains R-134a.
CH-5	SB-31	Gas detector	Solvents (alcohol, toluene) /Hydrocarbon	1. High sensitivity to solvents

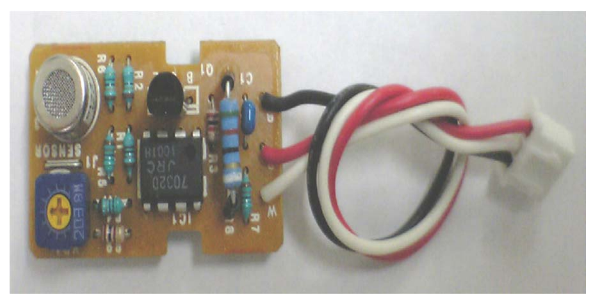


Fig. 5. An example of a SB-series model.

5. Results and conclusions

The database used in this study comprises a total of 25,679 measurements corresponding to 4 types of gas. The tests were performed with a 5x2 cross validation to statistically validate the difference between the accuracy rate for the different methods. Specifically, the objective was to confirm whether the use of MLP improved the results provided by independently used classifiers. In this case, the system would be capable of learning from the errors made by the different classifiers and could take them into account when making predictions. In order to perform the tests, the cases were grouped according to the total data, thus preventing the groups from having relevance in the testing phase.

Table 3 shows the results obtained for the different classifiers used during the problem by applying a 10-fold cross validation. The method producing the best results in this case was IBk.

By selecting the 9 classifiers with the best results and one of the

Table 3
Success rate for the different classifiers.

BayesNet	94.17423	NaiveBayes	73.71393	AdaBoostM1	49.6943
Bagging	99.40029	DecisionStump	49.6943	J48	99.52101
RandomForest	99.61837	IBk	99.60668	JRip	99.40808
LMT	99.17832	Logistic	90.52144	LogitBoost	97.07154
OneR	78.12999	SMO	79.6098	Stacking	25.20737
MultilayerPerceptron	96.97418	MultiClassClassifier	88.69894		

Table 4
Success applying a 5×2 cross validation with mixture.

Classifiers	I1	I1	I2	I2	I3	I3	I4	I4	I5	I5	Average
Bayes Net, Naive Bayes	12,751	12,757	12,758	12,763	12,764	12,763	12,751	12,769	12,758	12,768	12,760.2
Bagging, Naive Bayes	12,782	12,766	12,750	12,752	12,758	12,747	12,767	12,762	12,745	12,771	12,760.0
J48, Naive Bayes	12,772	12,780	12,784	12,758	12,786	12,750	12,773	12,783	12,772	12,767	12,772.5
Random Forest, Naive Bayes	12,797	12,785	12,774	12,783	12,780	12,779	12,787	12,779	12,792	12,796	12,785.2
JRIP, Naive Bayes	12,761	12,597	11,918	12,766	12,668	12,694	12,726	12,762	12,468	12,767	12,612.7
IBK, Naive Bayes	12,805	12,795	12,799	12,798	12,801	12,794	12,790	12,796	12,777	12,797	12,795.2
LMT, Naive Bayes	12,754	12,779	12,799	12,732	12,801	12,615	12,763	12,774	12,786	12,782	12,758.5
Logistic, Naive Bayes	12,788	12,788	12,771	12,766	12,774	12,792	12,780	12,785	12,719	12,755	12,771.8
Logit Boot, Naive Bayes	12,776	12,691	12,676	12,743	12,786	12,753	12,748	12,771	12,759	12,766	12,746.9
Naive Bayes, One R	12,782	12,066	12,765	12,717	12,437	12,651	12,385	12,415	11,292	12,656	12,416.6

Table 5
Success applying a 5×2 cross validation without mixture.

Classifier	I1	I1	I2	I2	I3	I3	I4	I4	I5	I5	Average
Bayes Net	12,103	12,062	12,064	12,048	12,053	12,112	12,075	12,072	12,075	12,083	12,074.7
Bagging	12,744	12,739	12,728	12,721	12,736	12,728	12,728	12,734	12,713	12,736	12,730.7
J48	12,744	12,750	12,746	12,727	12,760	12,722	12,744	12,750	12,743	12,737	12,742.3
Random Forest	12,773	12,760	12,761	12,754	12,764	12,761	12,750	12,759	12,759	12,760	12,760.1
JRIP	12,732	12,714	12,697	12,725	12,699	12,712	12,692	12,705	12,714	12,748	12,713.8
IBK	12,768	12,755	12,759	12,767	12,767	12,763	12,750	12,758	12,745	12,762	12,759.4
LMT	12,727	12,739	12,754	12,686	12,757	12,305	12,724	12,739	12,756	12,756	12,694.3
Logistic	11,635	11,648	11,739	11,561	11,586	11,795	11,626	11,649	11,473	11,690	11,640.2
Logit Boot	12,451	12,407	12,393	12,456	12,459	12,384	12,418	12,373	12,431	12,393	12,416.5
Naive Bayes	9608	9533	9519	9605	9654	9760	9783	9543	9792	9213	9601

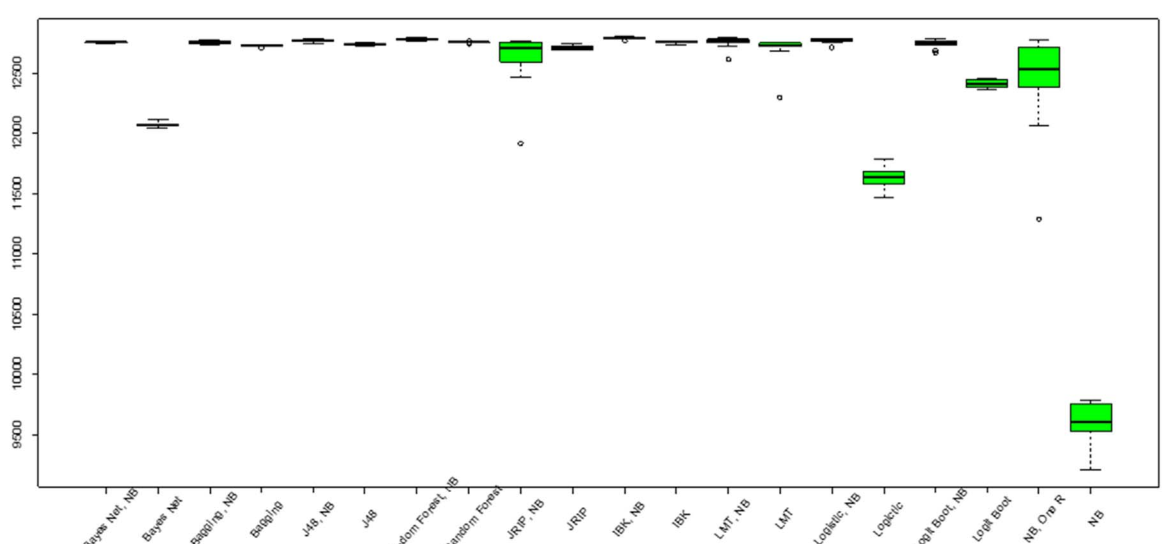


Fig. 6. Box plot with the information about the success.

other classifiers, and then applying a mixture of experts with an MLP having a learning rate of 0.45, moment 0.15, sigmoidal activation function and bias, the following results were produced, as shown in Table 4. The MLP has 4 output neurons, one for each class. The outputs are defined in the range [0.2, 0.8] for each class of odor, the minimum value means lack of odor, and 0.8 indicates that the odor is detected. According to this configuration, the function to minimize can be defined as the expression (6).

Table 5 shows the same information as Table 4 without mixture. All classifiers provide better results with the mixture, with the exception of JRIP. Moreover, we can see that the success rate is constant and greater than the success rate that we have with the MLP. Another

Table 6
Mann Whitney test with mixture and with out mixture.

Mixture	Classifier	Greater	Less	Two sided
Bayes Net, Naive Bayes	Bayes Net	0.0001	0.9999	0.0002
Bagging, Naive Bayes	Bagging	0.0001	0.9999	0.0002
J48, Naive Bayes	J48	0.0002	0.9998	0.0004
Random Forest, Naive Bayes	Random Forest	0.0001	0.9999	0.0002
JRIP, Naive Bayes	JRIP	0.5452	0.4849	0.9698
IBK, Naive Bayes	IBK	0.0001	0.9999	0.0002
LMT, Naive Bayes	LMT	0.0095	0.9923	0.0190
Logistic, Naive Bayes	Logistic	0.0001	0.9999	0.0002
Logit Boot, Naive Bayes	Logit Boot	0.0001	0.9999	0.0002
Naive Bayes, One R	Naive Bayes	0.0000	1.0000	0.0000

important issue is that methods such as Naïve Bayes obtain very good results with the mixture of One R, although they do not have high accuracy.

The information from Tables 4, 5 is represented in the box plot in Fig. 6. We can see in the image the distribution of the success with and without mixture.

Finally if we analyze the differences statistically using the information from Tables 4 and 5, we can apply Mann Whitney to determine the significance of the difference. The header represents the alternative hypothesis. Table 6 shows the results of the Mann Whitney test. The success with mixture is greater than without mixture for all methods except JRIP. With JRIP, we accept that both methods have the same distribution according to the information shown in Table 6.

The mixture of expert results in an increase in the success rate of classifiers, and is a good alternative to fusing the information of several experts. In this case an MLP was used to create the mixture and provided good results. The configuration of the MLP with the information of the sensors and the output of the classifiers allows us to learn about the misclassified elements; in this way, the MLP improves the results. In future works, it would be necessary to analyze the relevance of the number of classifiers used in the mixture in order to establish a relationship between the number of inputs associated with sensors and the inputs associated with the output of classifiers in the neural network. The main reason to use an MLP is due to the possibility of applying it in other cases studies; for example, fusion can be applied to prediction in time series. This issue will be analyzed with new cases studies.

Acknowledgement

This work has been supported by project Sociedades Humano-Agente en entornos Cloud Computing (Soha+C), SA213U13. Project co-financed with Junta Castilla y León funds.

References

[1] J.M. Corchad, J.F. De Paz, S. Rodrígu, J. Bajo, Model of experts for decision support in the diagnosis of leukemia patient, Artif. Intell. Med. (2009).

[2] S. Furao, T. Ogura, O. Hasegaw, An enhanced self-organizing incremental neural network for online unsupervised learning, Neural Netw. 20 (2007) 893–903.

[3] P. Boeker, On 'Electronic Nose' methodology, Sens. Actuators B: Chem. (2014) p2–17.

[4] M. Pardo, G. Niederjaufner, G. Benussi, E. Comini, G. Faglia, G. Sberveglieri, M. Holmberg, I. Lundstorm, Data preprocessing enhances the classification of different brands of Espresso coffee with an electronic nose, Sens. Actuators B 69 (2000) 397–403.

[5] J. Bajo, J.F. De Paz, G. Villarrubia, J.M. Corchado, Self-organizing architecture for information fusion in distributed sensor networks, Int. J. Distrib. Sens. Netw. 46 (3) (2015) 179–200.

[6] Wilson, A.D. (2011). Future applications of electronic-nose technologies in healthcare and biomedicine. wide spectra of quality control. IntechOpen.

[7] A. Campagnoli, L. Pinotti, G. Tognon, F. Cheli, A. Baldi, V. Dell'Orto, Potential application of electronic nose in processed animal proteins (PAP) detection in feedstuffs, Biotechnol. Agron. Soc. Environ. (2004) 253–255.

[8] Handbook of Machine Olfaction: electronic Nose Technology, in: T.C. Pearce, S.S. Schiffman, H.T. Nagle, J.W. Gardner (Eds.), , Wiley-VCH Verlag GmbH & Co, KGaA, Weinheim, 1994.

[9] Chen, H., Goubran, R.A., Mussivand, T. (2004). Improving the classification accuracy in electronic noses using multi-dimensional combining (MDC). IEEE.

[10] J.W. Gardner, P.N. Bartlett, A brief history of electronic noses, Sens. Actuators (1994) 211–220.

[11] N.F. Shilbayeh, M.Z. Iskandarani, Quality control of coffee using an electronic nose system, Am. J. Appl. Sci. (1994) 129–135.

[12] J. Rodriguez, C. Duran, A. Reyes, Electronic nose for quality control of colombian coffee through the detection of defects in "Cup Tests", Sensors (2010) 2010.

[13] H. Yu, J. Wang, H. Xiao, M. Liu, Quality grade identification of green tea using the eigenvalues of PCA based on the E-nose signals, Sens. Actuators B 140 (2009) 378–382.

[14] N.F. Shilbayeh, M.Z. Iskandarani, Quality control of coffee using an electronic nose system, Am. J. Appl. Sci. 1 (2) (2004) 129–135.

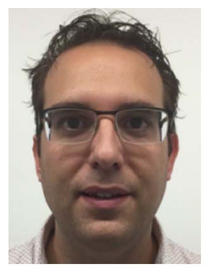
[15] A.Z. Berna, J. Lammertyn, S. Saevels, C.D. Natale, B.,M. Nicolai, Electronic nose systems to study shelf life and cultivar effect on tomato aroma profile, Sens. Actuators B 97 (2004) 324–333.

[16] Z. Ali, W.T. O'Hare, B.J. Theaker, Detection of bacterial contaminated milk by means of a quartz crystal microbalance based electronic nose, J. Therm. Anal. Calorim. 71 (2003) 155–161.

[17] R. Dutta, E.L. Hines, J.W. Gardner, P. Boilot, Bacteria classification using Cyranose 320 electronic nose, BioMedical Engineering Online, 2002.

[18] H.Chen, R.A. Goubran, T. Mussivand, Improving the classification accuracy in electronic noses using multi-dimensional combining (MDC), IEEE, 2004.

[19] Sensores-inistec, FIS, ⟨http://www.insistec.net/files/2_FIS.pdf⟩



Gabriel Villarrubia González. PhD Student on Computer Engineer. Currently, he is Assistant Professor at the Department of Computer Science at the University of Salamanca and member of the BISITE Research group. He obtained a Technical Engineering in Management of Computer in 2009 at the Pontifical University of Salamanca, then an Engineering in Computer Sciences degree in 2010 at the same university and postgraduate in Intelligent Systems at the University of Salamanca in 2011. He is co-author of more than 50 papers published in recognized journal, workshops and symposiums, most of them related to Intelligent Systems, Ambient Intelligence, and Distributed Systems. He has been member of the organizing and scientific committee of several international symposiums such as FUSION, PAAMS, MIS4TEL, ISAMI, PACBB.

Juan Francisco De Paz (PhD.). Received a PhD in Computer Science from the University of Salamanca (Spain) in 2010. He is Assistant Professor at the University of Salamanca and researcher at the BISITE research group (http://bisite.usal.es). He obtained a Technical Engineering in Systems Computer Sciences degree in 2003, an Engineering in Computer Sciences degree in 2005 at the University of Salamanca and Statistic degree in 2007 in the same University. He has been co-author of published papers in several journals, workshops and symposiums.

Dechen Pelki (PhD student). Phd student in Computer Science from the Polytechnic University of Madrid, Artificial Intelligence department. Bachelor of Engineering (BE), Information Technology 2014.

Fernando de la Prieta. PhD on Computer Engineer from the University of Salamanca. Currently, he is Assistant Professor at the Department of Computer Science of the same University and member of the BISITE Research group. He has been member of the organizing and scientific committee of several international symposiums such as ACM-SAC, FUSION, PAAMS, MIS4TEL, etc. and co-author of more than 60 papers published in recognized journal, workshops and symposiums, most of them related to Intelligent Systems, Ambient Intelligence, Distributed Systems, Technology Enhanced Learning and Information Security.

Sigeru Omatu is Professor of at the Department of Electronics, Information and Communication, Faculty of Engineering, Osaka Institute of Technology, Osaka, Japan. He received his Ph.D. in Electronic Engineering from Osaka Prefecture University and joined the faculty at University of Tokushima in 1969. He was Professor of University of Tokushima in 1988 and Professor of Osaka Prefecture University in 1995. He has been Professor of Osaka Institute of Technology since 2010. His honors and awards include the Best Paper Awards for Distributed Parameter System Theory, IEE of Japan, 1991, for Intelligent Classification, JSME, 1995, for Coin and Bill Classification, SICE, Japan, 1995, for Intelligent Smell Classification, IARIA, 2008, for Neuro-Control, IARIA, 2009. Furthermore, he received Ichimura Distinguished Award for Intelligent Classification, New Technology Development Foundation, 1996, SICE Fellow and IEEJ Outstanding Achievement Award, 2005. He used to be the President of ISCIA (Institute of Systems, Control and Information Association, Japan) in 2007 and President of Electronics, Information and Systems Society, IEEJ, 2004–2006, Regional Editor of EAAI. (IFAC Journal) for 1998–2001, Associate Editor of IEEE Transactions on Neural Networks for 1993–2003, and IMA Journal of Mathematical Control and Information (Oxford University Press) since 1991. His research area covers intelligent signal processing, pattern recognition, intelligent control, and adaptive control.