International Journal of Engineering Education Vol. 30, No. 3, pp. 533–542, 2014
Printed in Great Britain

0949-149X/91 $3.00+0.00

Emerging Computational Tools: Impact on Engineering Education and Computer Science Learning*

CORNELIO YÁÑEZ-MÁRQUEZ
Instituto Politécnico Nacional—Centro de Investigación en Computación IPN-CIC, Av. Juan de Dios Bátiz, Esq. Miguel Othón de Mendizábal, CIC Building, Mexico City 07738, Mexico. E-mail: cyanez@cic.ipn.mx

MARIO ALDAPE-PÉREZ, ITZAMÁ LÓPEZ-YÁÑEZ and OSCAR CAMACHO-NIETO
Instituto Politécnico Nacional—Centro de Innovación y Desarrollo Tecnológico en Cómputo IPN-CIDETEC, Av. Juan de Dios Bátiz, Esq. Miguel Othón de Mendizábal, CIDETEC Building, Mexico City 07738, Mexico. E-mail: maldape@gmail.com, itzama@gmail.com, oscarc@cic.ipn.mx

Computational tools have become an almost permanent element and one that must be present in novel schemes of interactive learning, practically for every educational topic in engineering education. A pretty natural environment for the application of such emerging technologies is that of scholar courses immersed in computer sciences learning, either in technical, superior or postgraduate teaching. Of particular interest are the teaching—learning processes of such topics as artificial intelligence, pattern recognition, neural networks, and associative memories. This interest arises from the magnitude of enrichment gleaned when the kind of emerging computational tools mentioned above are applied to these processes. The current paper focuses on describing and analyzing the positive impacts achieved in the teacher—student interactions through the application and everyday usage of a set of emerging computational tools that the authors have previously and currently employed in postgraduate computer sciences courses. Notice, however, that the results presented here are applicable—in a straightforward manner—to engineering education in general.

Keywords: emerging technologies; engineering education; computational tools; computer science learning

1. Introduction

Information technologies have shown an increasingly central role in recent engineering education systems. Thus, computational tools have become an almost permanent element and one that must be present in novel schemes of interactive learning, practically for every educational topic in engineering education [1]. This allows students to freely and constantly exchange experiences, content, and new ideas related to all subjects under study, thanks largely to distributed environments [2]. A pretty natural environment for the application of such emerging technologies is that of scholar courses immersed in computer sciences learning, either in technical, superior or postgraduate teaching [3].

The raw material in computer sciences is information and data in all its different meanings and varieties. In this sense, the birth of the Internet during the 1960s and of the World Wide Web in the late 1980s [4] have given rise to the symbiosis between information and communication technologies, which in turn has become a fundamental support to computer sciences; especially since the Semantic Web entered the scene [5].

Such kind of spectacular technological advances have revolutionized online engineering education systems, making "exchange" the key word in new educational paradigms throughout engineering [3]. Thus, content and knowledge management strategies in engineering education systems have become critical processes for any learning environment in engineering, and more specifically for computer sciences education [6–8].

Then, it is not surprising that information technologies have a great impact on teaching-learning processes in computer sciences, as well as other exact sciences such as mathematics [9–10]. This influence has been so pervasive that it has even given rise to a strong debate between specialist: on one hand, there are those who favor information technologies as a technical and functional expression of modernity (the Accolatory), and on the other hand (the Dismissive) there are those who protest the use and abuse of information technologies [11].

Independently from this debate, it is publicly known that both students and mentors working on different fields of computer sciences make everyday use of concepts, tools and procedures related to information technologies, in order to better acquire competences regarding diverse topics, both on software as well as hardware [12].

Of particular interest are the teaching—learning processes of such topics as artificial intelligence, pattern recognition, neural networks, and associative memories [13]. The current paper focuses on describing and analyzing the positive impacts achieved in the teacher–student interactions through

* Accepted 20 July 2013.

the application and everyday usage of a set of emerging computational tools and Technology Enhanced Learning that the authors have previously and currently employed in postgraduate computer sciences courses. It is noteworthy, however, that the results presented here are applicable—in a straightforward manner and without much modification—to engineering education in general [14].

The huge breadth of resources taken from information technologies to be applied to engineering education systems and computer sciences teaching is certainly impressive, making an exhaustive study beyond our scope. Therefore, the current paper presents only some examples which evidence how relevant the symbiosis between information technologies and computer sciences through the use of emerging computational tools and Technology Enhanced Learning is [6].

More specifically, this work deals with the use of two excellent tools taken from information technologies: WEKA [15] and MATLAB [16] (with its open source counterpart Octave [17]), which are shown through a case study related to the teaching the applications of computer sciences to the solution of environmental pollution problems [18–19]. Also, emerging computational tools related to educational content development and managing platforms (such as Moodle [20]), and to content distribution resources (such as Wikis [21]), are presented.

The rest of the paper is organized as follows: the chosen emerging computational tools are described in section 2; sections 2.1 to 2.4 present some elements of WEKA, MATLAB (and its open source counterpart Octave), Moodle, and Wikis, respectively. Section 3 includes the discussion of the main topic of this work and describes an example of the impact these emerging computational tools (particularly WEKA and MATLAB) have on such an important problem for engineering education systems such as the application of artificial intelligence to the prediction of environmental. Section 4 is dedicated to conclusions and future work, while bibliographical references and the author biographies are included at the end.

2. Presentation

This section is dedicated to presenting and describing briefly the emerging computational tools discussed in the current paper. The first subsection explores a key technology in recent years for worldwide pattern classifiers scientific research, which in turn is immersed in such fields of computer sciences as machine learning, pattern recognition, and data mining; this emerging computational tool is the WEKA open source software platform.

Later, subsection 2.2 refers to two quite similar tools: the MATLAB programming language and interactive environment, as well as its open source counterpart, Octave. Another must tool is the open source e–learning platform Moodle, which provides the added value of offering their integrated use in the learning environment, and is presented in the subsection 2.3. Finally, a short description of Wikis—websites which allow its users to add, modify, or delete its content via a web browser usually, and enables communities to write documents collaboratively, using a simple markup language and a web browser—is included.

2.1 WEKA

According to [22, 23], WEKA is a collection of machine learning algorithms for data mining tasks, including data pre–processing, classification, regression, clustering, association rules, and visualization. The algorithms can be applied directly to a dataset or called from a piece of Java code. WEKA is also well-suited for developing new machine learning schemes.

This superb open source tool was developed by a team of researchers at the University of Waikato, in New Zealand, and led by Ian H. Witten and Eibe Frank. As an acronym, WEKA stands for ''Waikato Environment for Knowledge Analysis''. This software is written in Java, which gives it very good portability, enabling it to run on the three mainstream desktop operating systems: Windows, Linux, and Mac. WEKA is distributed under the GNU license, and may be freely downloaded from the following URL: http://www.cs.waikato.ac.nz/ml/weka/

As can be seen in Fig. 1 (which shows its main interface), WEKA is made up by four main applications (Explorer, Experimenter, Knowledge Flow, and Simple CLI), which help the researcher to test different classification, clustering, or regression algorithms of any data set. Figure 2 includes a snapshot of the Explorer GUI.

Fig. 1. Main screen of WEKA.

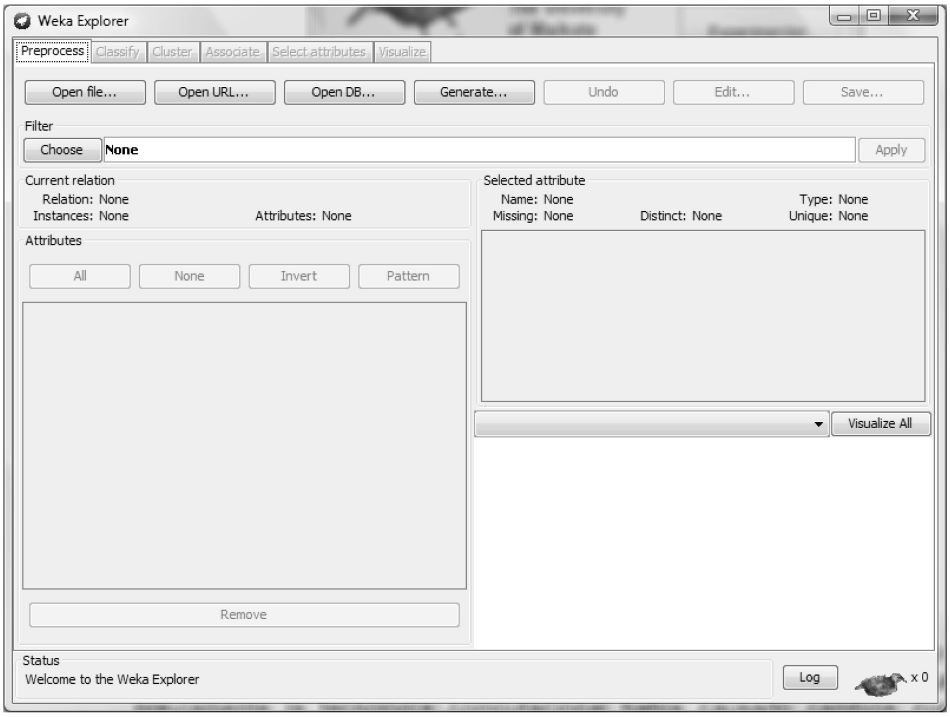

Fig. 2. Explorer—Module of WEKA.

By using the Explorer interface, the user is able to load a specific data set and apply to it a wide variety of artificial intelligence algorithms included in the platform, in order to tackle some problem from any field of human knowledge, including for instance environmental pollution.

One of the most critical tasks of data mining is surely deciding which of the available attributes of a data set are most useful to obtain the desired result. To this end, the WEKA Explorer has a dedicated panel for attribute selection—known as "Select attributes"—which enable the user to perform several algorithms and evaluation criteria on the selected data set, in order to identify the most important attributes. Given the large amount of combinations among different methods and different evaluation criteria, it is possible to configure a wide range of candidate techniques. Robustness of the selected attribute set can be tested via a cross-validation-based approach [15].

2.2 MATLAB and its open source alternative: octave

MATLAB is a programming platform focused on numerical computation and easy manipulation of vectors and matrices, besides offering visualization environments in two and three dimensions [16]. The great success achieved by MATLAB (particularly in research and engineering fields) is largely due to the ease with which a beginner can learn to use a programming environment, with little effort. The word MATLAB is actually an acronym, made up by the first letters in the words MATrix and LABoratory.

Regarding its ease of use, MATLAB offers engineers, scientists, and mathematicians an intuitive language for expressing complex problems and their solutions in a mathematical and graphical manner. This is done by integrating computation, programming, and visualization in a flexible, open and extensible environment. Thus complex numeric and symbolic problems can be solved in a fraction of the time required with other programming languages such as C, Fortran, or Java, even though MATLAB shares some syntax and even some implementation details with such programming languages as C or Java [24].

Another attractive aspect of MATLAB is its offer of toolboxes, which are collections of algorithms, functions, data structures and modules, packed as components, which are reared at solving specific problems of a particular field. Thus, there are toolboxes for image processing, for signal processing, for curve fitting, for bioinformatics, or even an aerospace toolbox.

Unlike WEKA, MATLAB is not open source; on the contrary: it is a very useful and versatile tool, yet quite expensive. It is developed and distributed by MathWorks, whose web site is: http://www.mathworks.com/products/matlab/.

Fortunately, there is an open source counterpart to MATLAB: it is the Octave platform [17], which uses a language compatible with MATLAB. Ori-

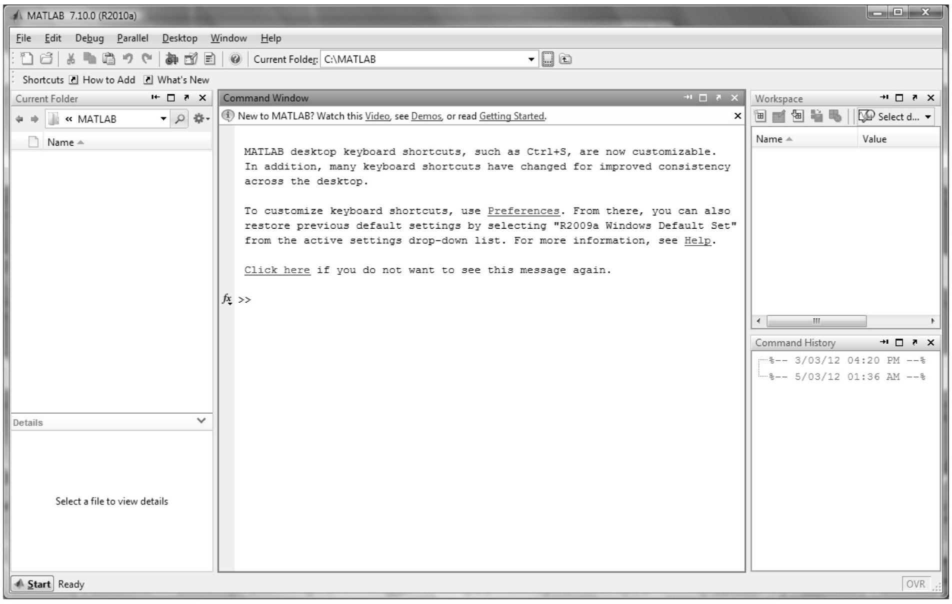

Fig. 3. Main screen of MATLAB.

ginally devised by James B. Rawlings from of the University of Wisconsin-Madison and John G. Ekerdt of the University of Texas, Octave has been developed using C and C++ under the GNU General Public License, making it freely distributable, and available for download from: http://www.octave.org/download.html. Figure 3 depicts a screenshot of MATLAB main screen, while the workshop of Octave is shown in Fig. 4.

2.3 *MOODLE*

The main web page of the moodle.org site contains a very clear and concise explanation of what Moodle was created for. There we can textually read: "Welcome to the Moodle community! Moodle is a Course Management System (CMS), also known as a Learning Management System (LMS) or a Virtual Learning Environment (VLE). It is a free web application that educators can use to create effective

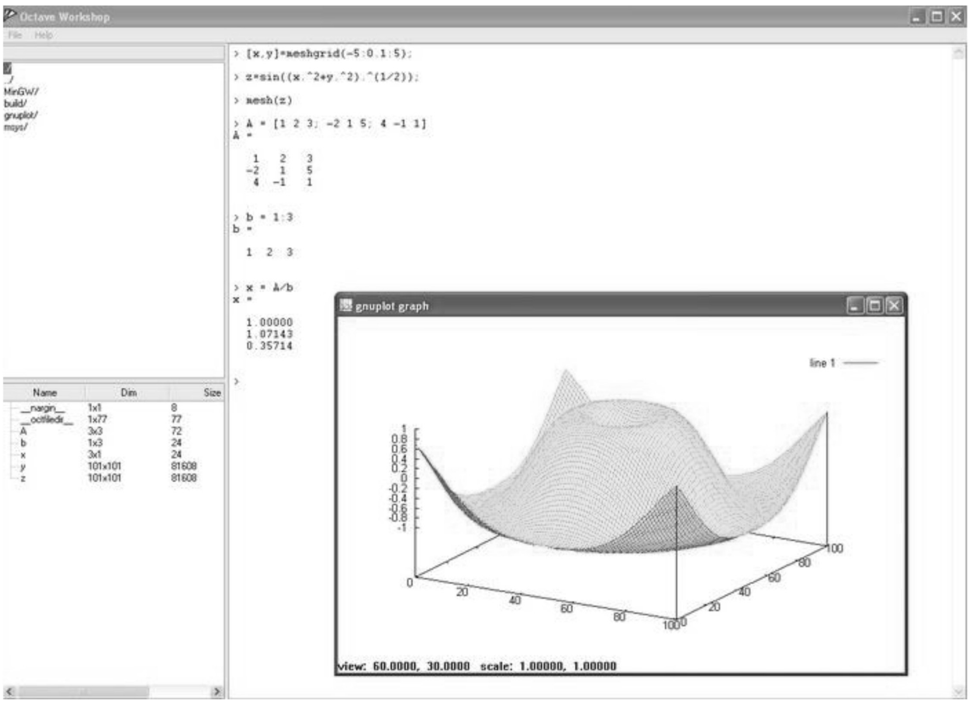

Fig. 4. Octave workshop.

Fig. 5. Main web page of the moodle.org site.

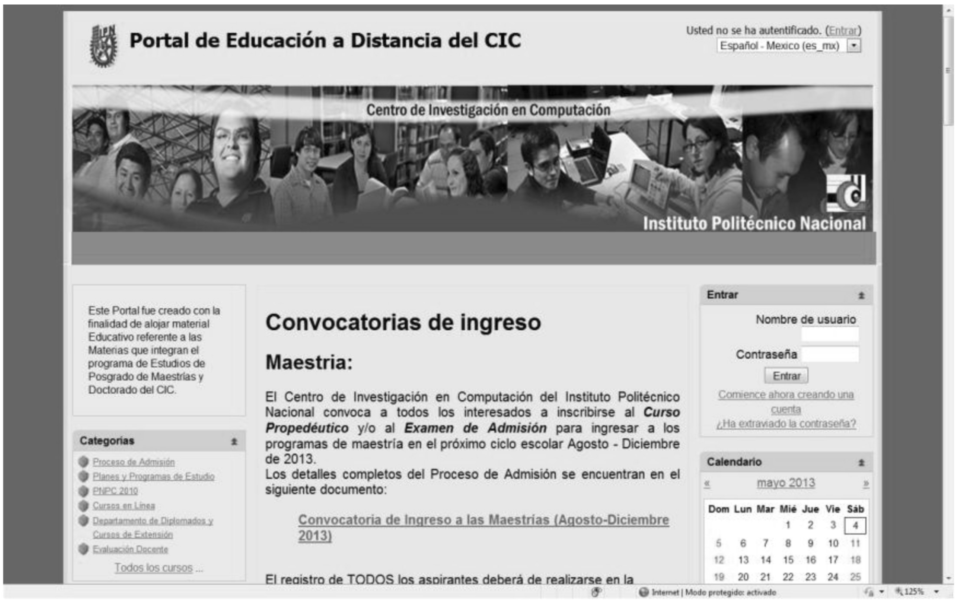

Fig. 6. Starting web page for the Moodle application used for educative management at the Center for Computing Research of the National Polytechnics Institute (www.cic.ipn.mx), in Mexico.

online learning sites. Moodle.org is our community site where Moodle is made and discussed.''

The fundamental pedagogic cornerstone of Moodle is the social constructionist learning philosophy, which is based on how humans build knowledge by interacting with each other and with learning materials, but not on our own: rather we do it in a social way. In this sense, Moodle enables the instructor to integrate into a course such contents as static learning material, interactive activities, or even social features, helping students to achieve their learning potential [25–26].

2.4 WIKIS

According to [27], the expression ''wiki'' is a contraction of the Hawaiian word ''wiki–wiki'' which means ''quick'' or ''fast''. When considering it as an emerging computational tool, a wiki can be defined as a ''collaborative web space where anyone can add content and anyone can edit content that has already been published''.

Wikis offer a shared online environment where students can actively participate in the collaborative creation and integration of knowledge, and thus they may be used to encourage the flourishing of dynamic online learning communities, where students meet to achieve a common goal. Thus, the members of such a Wiki community use this joint space to write, discuss, comment, edit, reflect, and evaluate their ideas and work, aiming at completing a shared outcome [32]. This forces educators to give more room to their students, particularly in this quintessential example of the so–called "age of participation", where rather than leading their pupils they need to coach said pupils in their ability to collaborate and share information, knowledge, or learning.

Although Wikis realize their promised potential when promoting community, collaboration, and interactivity, their application in the class rooms is still relatively sparse. One of the reasons for this late adoption of such a potentially wonderful tool is exactly its high degree of interactivity, collaboration, and freedom, leaving instructors liable to deem this new platform as unmanageable or just too complicated for their purposes. [33].

In [31], the authors provide valuable information for beginning the use of Wikis. There, it is explained that Wiki software can be described as coming in one of two flavors: Wiki only on one hand, with a minimal feature set (minimal registration process, history, page locking, and IP blocking). MediaWiki, the wiki software used for Wikipedia (http://wikipedia.org), is wiki-only. On the other hand there is the option with a full-featured management set. This includes challenged registration (visitors have to confirm their email address before modifying pages), workflow management, user permissions, image and file galleries, surveys, and a full administrative panel, plus more. TikiWiki, the wiki software used for WikiWackyWorld (http://www.wikiwackyworld.com), is a full-featured management set. Installation requires uploading or copying files to the server, creating a MySQL database and user account, and running the installation scripts. Installation packages are available for some wikis.

3. Discussion and example

The impact of emerging computational tools on engineering education and computer science learning is beyond doubt. Day to day mentors, teachers, researchers, tutors, scientists, and pedagogues witness the ubiquity of these tools.

These emerging computational tools have revolutionized the scope of educative and research processes. Class rooms, research and learning labs, field works, and interdisciplinary sessions with collaborative study evidence the breadth and depth of novel didactic resources which those in charge of educating new generations can take advantage of, both for engineering education in general, and computer science learning in particular.

The way to research in class rooms and labs of such fields as machine learning, artificial intelligence, pattern recognition, neural networks, and associative memories has been revamped by such tools as WEKA and MATLAB, while Moodle and Wikis have greatly transformed the organization of task like content, knowledge management, and data security [28–30]. For instance, all processes associated to managing the admission of new graduate students at the Center for Computing Research of the National Polytechnics Institute in Mexico (CIC IPN, http://www.cic.ipn.mx) are based on a Moodle application, designed and tuned specifically for educative administration (Fig. 6). Below is an example of the impact shown by the use of WEKA coupled with MATLAB for scientific research related to atmospheric pollutants forecasting in Mexico City.

3.1 Atmospheric pollution in Mexico City

The content of this subsection is strongly based on [34]. Environmental topics have gained the attention of increasingly large portions of global population. In different languages and through diverse means, civil associations launch campaigns for people to realize the importance of protecting the environment, even attracting the active participation of several governments.

Related to this increased awareness, Mexico City has been suffering grave problems derived from atmospheric pollution in recent decades; so grave in fact that the government has been forced to intervene. Thus arose the Mexico City Atmospheric Monitoring System (*Sistema de Monitoreo Atmosférico*, SIMAT in Spanish), which is the government body tasked with watching and monitoring the air quality in Mexico City.

SIMAT is committed to operate and maintain a trustworthy system for the monitoring of air quality in Mexico City, as well as analyzing and publishing this information in order to fulfill the current requirements and legislation. The objective of SIMAT is to watch and evaluate the air quality in Mexico City, as a pre-emptive measure for health protection of its inhabitants, in order to promptly inform the populace as well as enable decision making in prevention and air quality improvement programs. SIMAT is made up by four specialized subsystems, one Atmospheric Monitoring Mobile Unit, and a Calibration Standards Transfer Laboratory. One of these subsystems making up SIMAT is the RAMA (Automatic Atmospheric Monitoring Network, *Red Automática de Monitoreo Atmosférico* in Spanish), which takes continuous and permanent measurements of several contaminants: ozone (O_3), sulphur dioxide (SO_2), nitrous oxides (NO_x), carbon monoxide (CO), particulate matter less than 10 microns in diameter (PM_{10}), and particulate matter less than 2.5 microns in diameter ($PM_{2.5}$). Each measurement is taken automatically every hour.

In turn, the Air Quality Metropolitan Index (*Índice Metropolitano de la Calidad del Aire*, IMECA in Spanish) is a reference value for people to be aware of the pollution levels prevalent in any zone, in a precise and timely manner, in order to take appropriate protection measures. When the IMECA of any pollutant is greater than 100 points, its concentration is dangerous for health and, as the value of IMECA grows, the symptoms worsen, as can be seen in Table 1.

3.2 The use of WEKA and MATLAB (octave) in predicting some atmospheric pollutants and IMECA

Among the different activities undertaken by the government of the Federal District to tackle the problems caused by atmospheric pollution, there are the projects developed under the substantive program titled *Ciudad Sostenible* (Sustainable City), financed and managed by the Federal District Institute of Science and Technology (*Instituto de Ciencia y Tecnología del Distrito Federal* in Spanish). Notice that there are several other substantive programs similar to the former, such as: *Ciudad Saludable* (Healthy City), *Ciudad con Conectividad y Tecnología* (City with Connectivity and Technology), and *Ciudad con Industria Competitiva* (City with Competitive Industry).

Some researchers at CIC IPN have accepted the challenge posed by these projects, taking advantage of this valuable opportunity to apply several emerging computational tools to the teaching of computer sciences, particularly in areas such as artificial intelligence, machine learning, and data mining. Thus, students use WEKA as well as MATLAB or Octave for the prediction of some contaminants of interest and their IMECA, under the projects PIUTE10-77—Sensor inteligente para el monitoreo de los niveles de emisión de contaminantes en un vehículo automotor—and PICSO10-85—Aplicación de la inteligencia artificial para la predicción de contaminantes y del índice metropolitano de la calidad del aire de la Ciudad de México (IMECA).

For the experimental phase, data taken from the RAMA public database were used, especially those related to carbon monoxide (CO). The learning set for the experiments was built by taking the CO samples obtained at the Iztacalco sampling station, measured in parts per million (ppm) during the whole year 2010. Then, input patterns were generated by concatenating 10 consecutive samples, considering the output patterns as the following sample. These patterns make up a fundamental set of 8749 associations, which were presented for the learning phase to the Gamma Classifier, which is an original model of our research group and whose algorithm was implemented using MATLAB and Octave.

The test set was then built by generating associations of input/output patterns like those of the fundamental set, but using the samples measured by the same station during February 2011 as basis. Thus, the test set is made up by 661 associations.

The experimental results shown here predict the contaminant of interest, CO in this case; however, given that the concentration value of a contaminant is used to compute its corresponding IMECA (according to the previous explanation regarding SIMAT and IMECA), such predicted concentration of CO is of great importance for the prediction of the IMECA value.

For instance: on February 2, at 19:00 the proposed system predicted a CO concentration of 0.8 ppm, being the original value 0.8 ppm; as expected, the corresponding IMECA values are also equal: 7, equivalent to a Good condition. On the other hand, on February 16 at 13:00 the predicted and actual CO concentrations are 1.7 ppm and 1.4 ppm, respectively, while the corresponding IMECA values are 15 and 13, which give a Good condition for both. Thus, even though the prediction was not exact for the CO concentration, the IMECA level was correctly predicted: Good. Finally, on February 23 at 6:00 the predicted concentration for CO was 0.6 ppm, while the sampled value was 1.8 ppm, thus

Table 1. IMECA and its implications for health

IMECA	Condition	Effects on Health
0–50: green	Good	Suitable for conducting outdoor activities.
51–100: yellow	Regular	Possible discomfort in children, the elderly and people with illnesses.
101–150: orange	Bad	Cause of adverse health effects on the population, particularly on children and older adults with cardiovascular and/or respiratory illnesses such as asthma.
151–200: red	Very Bad	Cause of greater adverse health effects on the population, particularly on children and older adults with cardiovascular and/or respiratory illnesses such as asthma.
>200: purple	Extremely Bad	Cause of adverse health effects in the general population. Serious complications may present in children and older adults with cardiovascular and/or respiratory illnesses such as asthma.

Table 2. Example prediction results for CO concentration, IMECA value, and IMECA level. P stands for Predicted, while O means Original.

Sample	CO concentration (ppm)			IMECA value			IMECA level		
	P	O	Error	P	O	Error	P	O	Error
February 2 19:00	0.8	0.8	0.0	7	7	0	Good	Good	No
February 11 14:00	1.0	0.8	0.2	9	7	2	Good	Good	No
February 16 13:00	1.7	1.4	0.3	15	13	2	Good	Good	No
February 18 11:00	3.4	2.2	1.2	31	20	11	Good	Good	No
February 23 6:00	0.6	1.8	–1.2	5	16	–11	Good	Good	No
February 28 9:00	5.2	3.3	1.9	47	30	17	Good	Good	No

Table 3. Comparison of prediction performance for CO during February 2011.

	CO Concentration (ppm)		IMECA value (points)	
Method	RMSE	Bias	RMSE	Bias
IBk 1 (1-NN)	0.631270	–20.1	5.784323	–192
RBFNetwork	0.587618	–17.0	5.367577	–156
MultilayerPerceptron	0.575232	–44.8	5.254347	–431
IBk 3 (3-NN)	0.561241	–13.7	5.110281	–142
ConjunctiveRule	0.559703	10.1	5.106135	54
LinearRegression	0.556912	**0.3**	5.077463	–31
M5P	0.556749	–15.5	5.100206	–156
Gamma Classifier	0.556382	–4.6	5.068218	–25
SMOreg	0.554639	–14.8	**5.064037**	–165

rendering the IMECA values as 5 and 16, respectively. Although the difference between predicted concentrations is 1.2 ppm, making the difference in IMECA values of 11, the level for the index corresponds to the same condition for both predicted and actual value: Good in both instances.

Employing the same data, the WEKA platform was used to do a comparative study of the performance exhibited by the proposed method (the Gamma Classifier) against that of known methods, which are included in WEKA. Above are shown the performance results obtained by the Gamma Classifier as well as other methods from WEKA, when predicting CO concentration for each hour of the February month of 2011, using data taken during the whole 2010 year to learn.

The Rooted Mean Square Error (RMSE) is used as a performance comparison criteria, and is computed as shown below, when Pi is the i–th predicted value, and Oi is the i–th original (actual, sampled) value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - O_i)^2}$$

In order to determine how much the algorithm used is under- or overestimating the results, bias (which is computed as indicated below) is used.

$$Bias = \frac{1}{n}\sum_{i=1}^{n}(P_i - O_i)$$

As can be seen, the gamma Classifier exhibits the second best performance for CO concentration, being surpassed only by SMOreg (SVM based regression) on RMSE (0.5564 ppm and 0.5546 ppm respectively); and by LinearRegression on bias (–4.6 and 0.3 respectively).

On the other hand, regarding the IMECA value the situation improves: again the best method on RMSE is SMOreg and the second best is the Gamma Classifier, with errors of 5.0640 and 5.0682 points respectively; while the bias of the gamma Classifier is the lowest at –25 points.

Notice however that, although SMOreg has lower RMSE than the Gamma Classifier for both concentration and IMECA value, the difference is marginal: 0.001743 ppm for the first instance and 0.004181 points for the second.

On the other hand, SMOreg has a higher bias for both cases, especially for the IMECA value: the differences against the Gamma Classifier are 10.2 ppm and 140 points, respectively. Notice that the latter 140 points may represent a difference of almost two IMECA levels. Also, the SMOreg is quite slow, noticeable slower than the other methods tested in WEKA.

Similarily, LinearRegression offers a better bias than the proposed model regarding concentration, with a bias difference of 4.9 ppm while the RMSE difference is as small as 0.00053 ppm (smaller even than the corresponding difference between SMOreg and the Gamma Classifier). With respect to IMECA value, LinearRegression has a very good bias too:

the second best, only greater than that of the Gamma Classifier by 6 points.

4. Conclusions and future work

In this paper, the use and impact of emerging computational tools has been illustrated through a case study related to the application of WEKA, MATLAB, and Octave, to solving problems of environmental pollution in Mexico City. More specifically, students employed these tools to improve their learning–teaching process in the field of computer sciences. By taking advantage of WEKA, MATLAB, and Octave, those students have applied different machine learning and artificial intelligence models to predict future values of the concentration of carbon monoxide (CO), which is a pollutant of interest to compute the IMECA at Mexico City. Once these pollutant concentrations were predicted, based on the data taken from the RAMA public database, the performances of the models were compared, showing experimentally the utility exhibited by the Gamma Classifier, which was developed by the authors research group.

Such competences as developing, implementing, testing, tuning, applying, and comparing different computation models, are at the heart of what a researcher in the area of computer sciences does day in and day out; the empowerment offered to students by the emerging computational tools presented here, enabling them to learn, develop, and exercise said competences from their education and formation stages as researchers, is of great relevance. The results presented here are applicable—in a straightforward manner and without much modification—to engineering education in general.

Acknowledgements—The authors would like to thank the Instituto Politécnico Nacional (Secretaría Académica, COFAA grant 20130801, SIP, CIC, CIDETEC), the CONACyT, and SNI for their economical support to develop this work.

References

1. M. A. Bowdon and R. G. Carpenter, *Higher Education, Emerging Technologies, and Community Partnerships*, Information Science Reference, USA, 2011, pp. 58–139.
2. K. Kirkland, *Computer science: notable research and discoveries, Facts On File*, Infobase Publishing, New York, 2010, pp. 122–125.
3. C. Dede, Emerging technologies and distributed learning, *American Journal of Distance Education*, **10**(2), 1996, pp. 4–36.
4. T. Berners-Lee and M. Fischetti, *Weaving the Web*, Harper Collins Publishers, San Francisco, 1999, pp. 177–210.
5. H. Alesso and C. F. Smith, *Thinking on the Web: Berners-Lee Lee, Gödel, and Turing*, John Wiley & Sons, New Jersey, 2006, pp. 219–240.
6. R. M. Bottino, Framing Technology Enhanced Learning Environments: Some Perspectives and Notions, *International Journal of Knowledge Society Research*, **3**(1), 2012, pp. 67–81.
7. I. Nonaka, The Knowledge-Creating Company, *Harvard Business Review*, Harvard, 2007, pp. 162–171.
8. L. D. McLean, *A review and a critique of Nonaka and Takeuchi's theory of organizational knowledge creation*, 2004, http://faculty.mu.edu.sa/public/uploads/, accessed 29 March 2013.
9. A. A. Juan, M. A. Huertas, S. Trenholm and C. Steegmann, *Teaching mathematics online—emergent technologies and methodologies*, Information Science Reference, USA, 2012, pp. 259–383.
10. A. Kamakari and A. Drigas, Advanced E-Learning Services for Teachers, *International Journal of Knowledge Society Research*, **3**(4), 2012, pp. 85–96.
11. P. Moreno-Moreno and C. Yáñez-Márquez, *The New Informatics Technologies in Education Debate*, in M. D. Lytras et al. (Eds.): WSKS 2008, CCIS 19, Springer-Verlag, Berlin Heidelberg, 2008, pp. 291–296.
12. A. Durán, *De Diderot a Internet*, in Semanario de Investigación Científica del Diario La Jornada, México, D.F., 23 de agosto del 2001, pp. 1–3.
13. M. Iskander, *Innovative Techniques in Instruction Technology, E-learning, E-assessment, and Education*, Springer, New York, 2008, pp. 1–593.
14. I. E. Dror, Technology enhanced learning: The good, the bad, and the ugly, *Pragmatics & Cognition*, **16**(2), 2008, pp. 215–223.
15. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, *The WEKA Data Mining Software: An Update*, SIGKDD Explorations, **11**(1), 2009, pp. 10–18.
16. D. J. Higham, *MATLAB guide*, SIAM, Philadelphia, 2005, pp. 1–34.
17. A. Quarteroni and F. Saleri, *Scientific Computing with MATLAB and Octave*, Springer-Verlag, Berlin Heidelberg, 2006, pp. 1–302.
18. I. López-Yáñez, C. Yáñez-Márquez, O. Camacho-Nieto and A. J. Argüelles-Cruz, Prediction of air contaminant concentration based on an associative pattern classifier, *Revista Facultad de Ingeniería, Universidad de Antioquia*, **60**, 2011, pp. 411–418.
19. C. López-Martin, I. López-Yáñez and C. Yáñez-Márquez, Application of Gamma Classifier to Development Effort Prediction of Software Projects, *Applied Mathematics & Information Sciences*, **6**(3), 2012, pp. 411–418.
20. M. Antón-Rodríguez, M. A. Pérez-Juárez, F. J. Díaz-Pernas, F. J. Perozo-Rondón, M. Martínez-Zarzuela and D. González-Ortega , Moodle-Based Software to Support the Learning of Web Programming, *International Journal of Knowledge Society Research*, **3**(3), 2012, pp. 16–28.
21. M. A. Forment, X. De Pedro, M. J. Casañ, J. Piguillem and N. Galanis, Requirements for Successful Wikis in Collaborative Educational Scenarios, *International Journal of Knowledge Society Research*, **1**(3), 2010, pp. 44–58.
22. Weka 3: Data Mining Software in Java, http://www.cs.waikato.ac.nz/ml/weka/, accessed 30 March 2013.
23. I. H. Witten and E. Frank, *Data mining: practical machine learning tools and techniques*, Elsevier, Amsterdam, 2005, pp. 187–283.
24. T. A. Davies, *MATLAB Primer*, CRC Press, Boca Raton FL, 2011, pp. 1–18.
25. W. H. Rice IV, *Moodle: E-Learning Course Development*, Packt Publishing Ltd., Birmingham, UK., 2006, pp. 9–30.
26. J. Hollowell, *Moodle as a Curriculum and Information Management System*, Packt Publishing Ltd., Birmingham, UK., 2011, pp. 5–38.
27. W. Richardson, *Blogs, Wikis, Podcasts, and Other Powerful Web Tools for Classrooms*, SAGE Ltd., London, UK., 2010, pp. 55–68.
28. I. Tuomi, *From periphery to center: emerging research topics on knowledge society*, TEKES, Helsinki, Finland, 2001, pp. 41–49.
29. F. Stutzman, R. Capra and J. Thompson, Factors mediating disclosure in social network sites, *Computers in Human Behavior*, **27**(1), 2011, pp. 590–598.
30. P. Korovessis, Information Security Awareness in Academia, *International Journal of Knowledge Society Research*, **2**(4), 2011, pp. 1–17.

31. B. Huettner, M. K. Brown and C. James-Tanny, *Managing virtual teams: getting the most from wikis, blogs, and other collaborative tools*, Wordware Publishing, Inc., Texas, USA., 2007, pp. 277–284.
32. J. A. West and M. L. West, *Using Wikis for Online Collaboration*, Jossey-Bass, San Francisco, USA, 2009, pp. 21–78.
33. M. Barton and R. Cummings, *Wiki writing: collaborative learning in the college classroom*, University of Michigan Press, USA, 2011, pp. 177–236.
34. Secretaría del Medio Ambiente. Sistema de Monitoreo Atmosférico de la Ciudad de México. http://www.calidadaire.df.gob.mx/calidadaire/index.php, accessed 29 March 2013.

Cornelio Yáñez-Márquez obtained his Bachelor degree (1989) on Physics and Mathematics at National Polytechnics Institute (IPN) Physics and Mathematics Superior School. His MSc (1995) and PhD (2002) degrees were received at IPN Center for Computing Research (CIC). Currently a Researcher Professor, Titular C, at IPN CIC. He was granted the Lázaro Cárdenas Award by the President of the Republic. Member of the National Researchers System (SNI). Areas of interest: Associative Memories, Neural Networks, Mathematical Morphology, and Software Engineering.

Mario Aldape-Pérez received his B.E. degree in electrical engineering from the Autonomous Metropolitan University, Azcapotzalco, Mexico, in 2004, and the M.Sc. degree in computer engineering and the Ph.D. degree in computer science from the National Polytechnic Institute of Mexico, Mexico, in 2007 and 2011, respectively. In 2010, he joined the Department of Computer Systems Engineering from the School of Computing, National Polytechnic Institute of Mexico, as an Associate Professor, and in 2012 became Professor. His current research interests include Unconventional Computing, Associative Memories, Soft Computing and FPGA Implementation of High Performance Pattern Classification algorithms. Dr. Aldape-Pérez is a member of the IEEE Computer Society. He is currently the President of the Academy of Digital Systems at the School of Computing of the National Polytechnic Institute of Mexico.

Itzamá López-Yáñez received his Bachelor degree as Information Systems Engineer (2003) at Monterrey Institute of Technology and Superior Studies (ITESM), while the MSc (2007) and PhD (2011) degrees on Computer Sciences at National Polytechnics Institute (IPN) Center for Computing Research (CIC), both with mention of Honor. He was granted the Lázaro Cárdenas 2012 Award by the President of the Republic. Currently he is a Professor at IPN Center for Technological Design and Development in Computer Science (CIDETEC), and member of the IEEE Communications Society. Areas of interest: Associative Memories, Neural Networks, Software Engineering, and Pattern Classification, in particular the Gamma classifier.

Oscar Camacho Nieto obtained his Bachelor degree as Communications and electronics Engineer (1989) at National Polytechnics Institute (IPN) Superior School of Mechanical and Electrical Engineering (ESIME), while the MSc degree (1995) on Computer engineering and the PhD degree (2003) on Computer Sciences, both at IPN Center for Computing Research (CIC). Currently a Researcher Professor, Titular C, at IPN, and Director of the IPN Center for Technological Design and Development in Computer Science (CIDETEC). Areas of interest: Computer Architectures, Associative memories, Microprocessors, Digital Systems, and Neural Networks.