CHAPTER-22

PATTERN RECOGNITION TECHNIQUE IN E-LEARNING WITH SOFT COMPUTATION

[1]TIRBHUWAN TYAGI
[1]*Assistant Professor, HLM Group of Institutions*

Chapter ID: NSP/ICAAR-2023/A-22

ABSTRACT

In this article an innovative platform is being presented that integrates intelligent agents in E-learning environments. An approach to recognize automatically the learning styles of individual learners according to the actions that he or she has performed in an e-learning environment is introduced with the help of a scalable and interoperable integration platform supporting various assessment techniques for e-learning environments. These techniques are implemented in order to provide intelligent assessment services to computational intelligent pattern recognition technique is based upon soft computing.

Keywords: *E-learning Systems, learning styles, Artificial Neural Networks, Web-based instruction.*

INTRODUCTION

Learners have dissimilar orientation of learning such as some prefer graphics, like diagrams and blueprints, while others prefer written material, some are more comfortable with facts, data and experimentation and some may prefer principles and theories (Felder & Silvermann 1988).E-learning environments can take advantage of these different forms of learning by recognizing the style of each individual learner using the system and adapting the content of courses to match this style. There are a few systems that are actually capable of adapting courses' contents according to learners' learning styles (Carver et al. 1999; Paredes & Rodriguez 2002; Stash & Brau 2004). In these systems, the learning materials are then presented in the way that best fit the learning style of each learner, which is usually assessed through a predefined questionnaire. However, answering long questionnaires is a time-consuming task that learners are not always willing to carry out and, consequently, results become unreliable (Stash et al. 2004). This paper describes an approach to the problem of mapping learner's actions within E-learning environments into learning styles. The method is based on Artificial Neural Networks (ANNs). Neural networks are computational models for classification inspired by the neural structure of the brain; models that have proven to produce very accurate classifiers. In the proposed approach, feed-forward neural networks are used to recognize learners' learning styles based upon the actions they have performed in an e-learning system.

LEARNING STYLES

Learning styles characteristics as cognitive, affective and psychological behaviors that serve relatively stable indicators of how learners perceive, interact with and respond to learning environments. In the last decade, several learning styles models have been proposed. A review of learning styles, analyzing their reliability, validity and implication for pedagogy, can be found in (Coffield et al. 2004a;b).

The authors of this review concluded that in the field of learning styles, there is a lack of theoretical coherence and a common framework (Coffield et al. 2004b, p. 145).

Experimental research on the application of learning styles in computer-based education provides support for the view that learning can be enhanced through the presentation of materials that are consistent with a learners' particular learning style.

In this paper, the model suggested by Felder and Silverman (1988) for engineering education, which classifies students according to their position in several scales that evaluate how they perceive and process information. This model classifies learners according to four dimensions:

- **Perception:** What type of information does the student preferentially perceive: sensory (external) --sights, sounds, physical sensations, or intuitive (internal)--possibilities, insights, hunches?
  - **Sensory learners:** These like facts, data and experimentation. They perceive concrete, practical, and are oriented towards facts and procedural information. When solving problems, sensory learners are routinely very patient with details and usually dislike surprises. Because of these characteristics, they show a slower reaction to problems, but they typically present a better outcome.
  - **Intuitive learners:** intuitive learners prefer theories and principles. They rapidly become bored with details and mechanical problem solving. Innovation is what attracts intuitive learners' attention. They generally solve problems quickly, not paying much attention to details. This makes them fast but prone to errors and, then, they often get lower qualifications than sensitive learners.
- **Input:** Through which sensory channel is external information most effectively perceived: visual--pictures, diagrams, graphs, demonstrations or verbal--written or spoken sounds?
  - **Visual learners:** they remember, understand and assimilate information better if it is presented to them in a visual way. They tend to remember graphics, pictures, diagrams, time lines, blueprints, presentations and any other visual material.
  - **Verbal learners:** cognitive scientists have established that our brains generally convert written works into their spoken equivalents and process them in the same way that they process spoken words (Felder & Brent 2005). Hence, verbal learners are not only those who prefer auditory material but also those who remember well what they hear and what they read.
- **Processing:** How does the learner prefer to process information: actively--through engagement in physical activities or discussions, or reflectively--through introspection?
  - **Active learners:** they feel more comfortable with active experimentation than with reflexive observation. An active person learns by trying things out and working with others. They like doing something in the external world with the received information. Active learners work well in groups and in situations that require their participation.
  - **Reflective learners:** reflective learners prefer introspective examination and manipulation of information. They learn by thinking things through and working alone or with another person.
  - **Understanding**: How does the student progress towards understanding: sequentially in continual steps, or globally--in large jumps, holistically?
  - **Sequential learners:** sequential learners follow a line of reasoning when progressing towards the solution of a problem; they like things to be linear. They learn better if information is presented in a steady progression of complexity and difficulty (i.e. they learn in small incremental steps).
  - **Global learners:** global learners make intuitive leaps and may be unable to explain how they come up with solutions. They are holistic, system thinkers; they learn in large leaps. They

need to understand the whole before understanding the parts that compose it; they need to get the 'big picture'.

These four dimensions of learning allow to obtain 16 different learning styles (i.e. in the perception dimension, a student would either be sensitive or intuitive). However, each dimension can be rated on a scale. ANNs can be used to recognize the learning styles of learners based on the actions they perform in an e-learning system.

ANNs

The human brain is composed of cells (neurons) that are the only cells capable of communicating with each other. This is one of the capabilities that allow humans to exhibit intelligent behavior. ANNs are computational models based on the biological neural structure of the brain, as first proposed by McCulloch and Pitts (1943). This computational model, also known as connectionism, aims to mathematically represent and reproduce the way a human nervous system works.

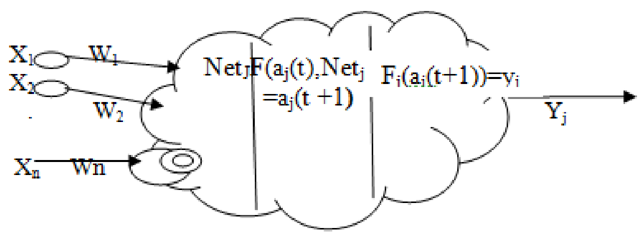

Fig 1. Main components of an artificial neuron

The neuron is the basic processing unit. Each neuron receives signals from others neurons, processes these signals and transmits an output to its neighbors'. Some of the signals it receives excite it while others inhibit it. Usually, an exciting signal is represented as a positive real value, while a negative value is assigned to an inhibiting signal. As shown in Fig 1, the Net input of a neuron is the sum of values that arrive at its input. This value represents the excitation level of the neuron. If it exceeds a certain threshold, the neuron fires an output signal to its neighbors. From this input–process–output model, neurons are classified with respect to where the signals come from. If the signals that the neuron receives come from the environment, it is called an input neuron. If it receives the input from other neurons and transmits its outcome to others as well, it is called a hidden neuron. Finally, if it sends its output to the environment, then it is called an output neuron. Usually, neurons sharing similar characteristics are grouped together, forming layers of neurons. What distinguishes one layer of neurons from another are their inputs and outputs. So, neurons belonging to layer i receive their input from layer i-1, and they send their output to layer i+1. Three kinds of layers are distinguished in ANN literature. If the layer contains input neurons, it is an input layer; if it contains hidden neurons, it is a hidden layer; and if it contains output neurons then it is an output layer.

An ANN is termed feed-forward, if neurons belonging to layer i receive input only from layer i-1, and only send output to layer i+1. This means that in feed forward neural networks, there are neither connection between neurons from the same layer (lateral connections) nor from previous layers (recurrent connections).Information in this class of networks only flows forward, from the input to the output passing through hidden layers.

Structure and Operation of ANNs

The fundamental unit of processing in ANNs is the Artificial Neuron (AN); see Fig 1. ANs are mathematical models that represent the general characteristics of biological neurons. These biological features are mathematically modeled and their contribution to the learning process of ANNs is explained as follows.

When a signal is transmitted from processing unit i to processing unit j, the signal x_i is modified by the synaptic weight (w_{ij}) associated with this communication channel. The modulated signals that arrive at unit j are added to form the net input Net_j as is shown in

$$Net_j = \sum_i x_i w_{ij} \tag{1}$$

Each neuron is characterized in any instant of time t by a real value, called activation state or activation level, $a_j(t)$ Also, there is a function F, called activation function, which determines the next activation state based on the current activation state and the Net_j input of the neuron.

$$F(a_j(t),Net_j) = a_j(t+1) \tag{2}$$

Associated with each unit there exists an output function, fj, that transforms the current activation level into an output signal y_i. This signal is sent through a unidirectional communication channel to other units in the network

$$F_j(a_j(t+1)) = y_j \tag{3}$$

LEARNING ALGORITHM

A learning algorithm is the process by which an ANN generates internal changes so that it can adapt its behavior in response to the environment. When there is an external agent involved in the learning process, it receives the name of supervised learning. Back-propagation is a supervised learning algorithm, used in feed-forward neural networks, which reduce the global error produced by the network over the weight space. BPN nets operate in two steps. In the first step, called training process, the network is initialized with random small values in its weights. The goal of this process is to find a set of weight values that minimize the global error of the network, given in equation (5). The second step is called generalization. In this step, the network has already learned an internal representation of the previously presented patterns and becomes able to classify novel patterns presented as inputs.

The learning process of a BPN is briefly described as follows: a pattern is a two-tuple $P_i = (X_i, T_i)$ where X_i is a set of values that will be presented as input to the network and T_i is a set of values that represent the desired target for the values presented at the input layer. For the network to learn a pattern, P_i, the values of the pattern has to be presented at the input layer first. These values are taken as stimulus and are propagated forward until the output layer is reached. In the output, a set (O) of values obtained by the network (called oj) are compared with the set of desired values (Ti) to obtain the pattern error according to

$$Err_i = \sum_{J=1}^{M} (aj-tj)^2, \tag{4}$$

where M is the number of neurons in the output layer, oi $\in$ O and tj $\in$ T. Hence, the global error of the network is calculated considering all patterns as follows:

$$Err = \frac{1}{2p}\sum_{K=1}^{P}\sum_{J=1}^{M}(oj^{(k)} - tj^{(k)})^2 , \tag{5}$$

where P is the number of patterns in the training set. This error is the one that the training procedure tries to minimize. These error values are back propagated through the network to adjust the values of its connection weights so that the change in values is proportional to the gradient descent of the error in equation (6). This weight adjustment rule is known as the Generalized Delta Rule (GDR) (Rumelhart et al. 1986;1988) and is a generalization of the Widrow– Hoff delta rule (Widrow & Hoff 1960)

$$\begin{aligned}\Delta W_{ij} &= \frac{dErr}{dWij} \\ &= \frac{d}{dWij}\,\frac{1}{2}\sum_{K=1}^{M}(o_k - t_k)^2 \\ &= \sum_{K=1}^{M}(o_k - t_k)\frac{dok}{dWij}\end{aligned} \tag{6}$$

This equation can also be rewritten as equation (7), with addition of the momentum (Rumelhart et al. 1986) term to achieve faster convergence. The momentum part of the equation, $\beta(W(t - 1) - W(t - 2))$, allows a network to respond not only to the local gradient but also to recent trends in the error surface.

$$W(t) = W(t - 1) - \alpha\delta_j y_j + \beta(W(t - 1) - W(t - 2)), \tag{7}$$

where δ_j is calculated according to the rule by

$$\delta_j = \begin{cases} f'(zj)(oi - tj) & (\text{output : layer}) \\ f'(zi)\sum_i W\,ij\delta j & (\text{hidden : layer}) \end{cases}$$

This process of weight adjustment is repeated until a desired error threshold is reached.

Modeling learning styles with feed-forward neural networks

The neural network architecture proposed in this paper aims to find a mapping between learner's actions in the system and the learning style that they best fit. To achieve this goal, inputs of the network must be identified to its outputs and the meaning of their possible values. It is also necessary to determine other architectural parameters, such as the number of hidden layers to be used, the number of processing units in each of the hidden layers, the activation function to be used in the processing units and the learning coefficient of the network. Each of these issues is analyzed in the following subsections.

INPUT LAYER

To represent the input of the network, we propose the use of one processing unit (neuron) in the input layer per observed action in the system. These actions are as follows:

- **Reading material:** academic units can be presented using both abstract (theories) and concrete material (exercises). What kind of material is the learner most interested in? Access to examples: in each academic unit, a number of examples are presented to learner. In relation to the total number of available examples, how many of them has the student accessed.

- **Answer changes:** Does the learner change the answers of the exam before he hands it over? If yes, what is the percentage of answers he has changed?
- Exercises: a number of exercises are also included in academic units. In relation to the total number of available exercises, how many exercises has the learner accessed to?
- **Exam delivery time:** each exam has an associated time –to solve. What is the relation between the learner's exam delivery time and the units' time –to solve?
- **Exam revision:** in relation to the time to solve of the exam, what was the percentage of time spent by the student checking the correctness of the exam?
- **Chat usage:** the student may ignore the chat, read other students' messages or read/write messages with others.
- **Forum usage:** the learner may ignore the forum, read other students posted messages or post messages in the forum.
- **Mail usage:** the learner may use (or not) the e-mail.
- **Information access:** information in academic units is presented following a line of reasoning. How has the learner followed that line of reasoning? Lineally, or has he or she visited a random sequence of items?

These values have to be encoded in the real interval [5;15] as expected by the neurons in the input layer of the network. This interval was intentionally selected to match the expected domain of the activation function selected for the units of the net as shown in subsection 'Network architecture and parameters'. Table 1 summarizes the input vector, X, representation.

OUTPUT LAYER

The output of the network should approximate the learning style of the students based on the actions

Table 1. Input Representation

x	Action	-5	+5
x0	Reading material	Abstract	Concrete
x1	Access to examples	Few	Much
x2	Answer changes	Few	Much
x3	Exercises	Few	Much
x4	Exam delivery time	Quick	Slow
x5	Exam revision	Few	Much
x6	Chat usage	Don't Use	Read & Write
x7	Forum usage	Don't Use	Read & Write
x8	Mail usage	Don't Use	Read & Write
x9	Information access	Lineal	Global

Table 2. Output Representation

O	Dimensions	-1	1
O0	Perception	Intuitive	Sensitive
O1	Processing	Active	Reflective
O2	Understanding	Sequential	Global

Presented at the input layer. In this case, we propose the use of one processing neuron in this layer per learning style dimension used in the model. In this work, only three of the four dimensions of the Felder–Silverman model have been used to model students' learning style. These dimensions are as follows:

- **Perception:** this dimension determines whether the style of the student is intuitive or sensitive.
- **Processing:** this dimension decides whether a student's leaning style better fits active or reflective.
- **Understanding:** this dimension informs whether the student learning style is sequential or global.
- Table 2 summarizes the output vector representation.

Hidden layer

In a multi-layer perceptron using continuous nonlinear hidden-layer activation functions, as proposed in this work, one hidden layer with an arbitrarily large number of units suffices for the 'universal approximation property' (Hornik et al. 1989; Bishop 1995). However, there is still no theory about how many hidden units are needed to approximate any given function. A total of 24 units have been empirically found as appropriate for the task at hand, considering that this layer has to have enough processing units to represent the nonlinear aspects of the model and not too many for making the training process very complex.

Network Architecture and Parameters

The network is trained using the GDR (Rumelhart et al. 1986;1988), a generalization of the Widrow–Hoff delta rule (Widrow & Hoff 1960). The activation function used in the network units is the hyperbolic tangent function shown in

$$f(Z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \quad (9)$$

Many interesting properties of this function make it suitable to represent the activation function of the network. On the one hand, the function domain can be restricted to [5;15] where the function reaches more than 99.99% of its range; the different observed aspects are then projected to this range. Another interesting property is that its derivate can be defined in terms of the function's output; this makes calculation easier as is shown in equation (10). A further important property of the hyperbolic tangent function is that its range is [1; 11]; this property can be utilized to represent the different ranges of the learning style dimensions.

$$\frac{d}{dx}\tanh(x) = \frac{d}{dx}\frac{e^{z} - e^{-z}}{e^{x} + e^{-x}}$$

$$= \frac{d}{dx}\frac{\sinh(x)}{\cosh(x)}$$

$$= \{\cosh^2(x) - \sinh^2(x) / \cosh^2(x)\} \quad (10)$$

$$= 1 - \tanh^2(x)$$

This architecture parameter, along with the proper selection of learning rates and momentum coefficients, defines the specific values selected for this work. The learning rate, ά, was set to a small value between 0.1and 0.25 so that the representation acquired can be a faithful one.

Table 3. Proposed Architectural Parameters

Parameter	Value
Number of Input Neurons	10
Number of Hidden Neurons	24
Number of Output Neurons	3
Activation Functions	Hyperbolic Tangent
Learning Rate	0.02
Momentum	0.5

To sum up, the proposed architecture for learning style recognition is a three-layered feed-forward neural network with Batch Gradient Descent with the Momentum learning algorithm. In this architecture, the first layer contains a total of 10 input neurons; the hidden layer contains 24 processing units that are connected to 3 output units in the third layer. Information about the network architecture and other parameters is summarized in Table 3.

EXPERIMENTAL RESULTS

Estimating the accuracy of classifiers induced by supervised learning algorithms (i.e. classifier's probability of correctly recognizing a randomly selected instance) is important not only to predict its future prediction accuracy but also for choosing a classifier from a given set (model selection) or combining classifiers (Wolpert 1992). In order to evaluate the proposed approach, an artificial dataset for experimentation by simulating the actions of learners is generated. For this task, it is considered that each learner has a particular learning style denoted by a set of preferred actions and behaves according to it. To determine the best number of processing units in the hidden layer, the network is trained by varying this parameter and estimated the architecture accuracy using k-fold cross-validation with k=10. Each of the 10 folds was composed of a set of 90 training patterns and 10 test patterns. To measure the accuracy of each classifier, the global error (equation (5)) is calculated and produces when it is stimulated with the test cases. Each of the output dimensions was considered independently in the calculation.

CONCLUSIONS

This paper described an approach based on feed-forward neural networks to infer the learning styles of learner automatically. The back-propagation algorithm to train the ANN is described in this work, in addition, a neural network architecture that learns the associations between students' actions in e-learning environments and their corresponding. The advantage of this approach is twofold. First, an automatic mechanism for style recognition facilitates the gathering of information about learning preferences, making it imperceptible to students. Second, the proposed algorithm uses the recent history of system usage so that systems using this approach can recognize changes in learning styles or some of their dimensions over time. The recognition mechanism described in this paper can be introduced in adaptive e-learning environments to help in the detection of learners' learning styles and, thus, conveniently adapt the contents of academic courses that are presented to them. It can also be extended to consider further input actions available in particular e-learning systems or domains

REFERENCES

1. *Ayre M. & Nafalski A. (2000) Recognising diverse learning styles in teaching and assessment of electronic engineering. In 30th ASEE/IEEE Frontiers in Education Conference (eds D. Budny & G. Bjedov), pp. 18–23.Kansas City, MO.*
2. *Budhu M. (2002) Interactive web-based learning using interactive multimedia simulations. International Conference on Engineering Education (ICEE 2002), pp.1–6.*
3. *Carver C.A.J., Howard R.A. & Lane W.D. (1999) Enhancing student learning through hypermedia courseware and incorporation of student learning styles. IEEE Transactions on Education 42, pp. 33–38.*
4. *Coffield F., Moseley D., Hall E. & Ecclestone K. (2004a) Learning styles and pedagogy in post-16 learning: a systematic and critical review. Technical Report 041543, Learning and Skills Research Centre.*
5. *Felder R.M. & Silvermann L.K. (1988) Learning and teaching styles in engineering education. Journal of Engineering Education 78, 674–681.*
6. *Hornik K., Stinchcombe M. & White H. (1989) Multilayer feed forward networks are universal approximators. Neural Networks 2, 359–366.*
7. *International Network for Engineering Education & Research, Manchester, UK.*
8. *J.E. Villaverde, D. Godoy (2006) Learning style's recognition with feed forword neural networks, journal of computer assisted learning 22,pp197-206*
9. *Kim K.S. & Moore J.L. (2005) Web-based learning: factors affecting students' satisfaction and learning experience.*
10. *Kuri N.P. & Truzzi O.M.S. (2002) Learning styles of freshmen engineering students. In International Conference on Engineering Education (ICEE 2002), Manchester, UK.*
11. *McCulloch W.S. & Pitts W. (1943) A logical calculus of ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 5, 115–133.*
12. *Pen˜a C.I., Marzo J.L. & de la Rosa J.L (2002) Intelligent agents in a teaching and learning environment on the Web. In ICALT2002. IEEE, Kazan, Russia.*
13. *Rumelhart D.E., Hinton G.E. & Williams R.J. (1986) Learning representations by back-propagating errors. Nature 323, 533–536.*
14. *Stash N. & Brau P.D. (2004) Incorporating cognitive styles in AHA! (The adaptive hypermedia architecture). In IASTED International Conference (ed. V. Uskov), pp. 378–383. ACTA Press, Innsbruck, Austria.*