
GPNN Techniques in Learning Assessment Systems

John Vrettaros*, John Pavlopoulos,
Athanasios S. Drigas*

Institute of Informatics & Telecommunications, Net Media Lab
N.C.S.R. Demokritos
Terma Patriarchou Grigoriou & Neapoleos 27, Agia Paraskevi, 153 10,
Athens, Greece
E-mail: jvr@iit.demokritos.gr
E-mail: annis.pavlo@gmail.com
E-mail: dr@iit.demokritos.gr
*Corresponding author

George Vouros

Aegean University, Info and Communication Systems Eng,
83200, Karlovassi, Samos, Greece
E-mail: georgev@aegean.gr

Abstract: The goal of this study is the development of an assessment system with the support of a Neural Network approach optimized with the use of Genetic Programming. The data used as training data are real data derived from an educational project. The developed system is able to assess learners' answers through various criteria and has been proved capable of assessing data from both single select and multiple choice questions in an e-learning environment.

Keywords: neural network, genetic programming, assessment system, learners, e-learning, GPNN, expert system.

Reference to this paper should be made as follows: Vrettaros, J., Pavlopoulos, J., Vouros, G. and Drigas, A. (2008) 'GPNN Techniques in Learning Assessment Systems', *Int. J. of ...*, Vol. x, No. x. pp. 000-000

Biographical notes:

John Vrettaros (M.Sc.) is a professor of physics and informatics and a pedagogist. He is an associate researcher at the Institute of Informatics & Telecommunications of N.C.S.R. Demokritos. He is the coordinator of e-learning projects within Net Media Lab and he has published more than 40 international & national articles in ICTs, 4 books, 25 educational CD-Roms, & several patents.

John Pavlopoulos is an Electrical and Computer Engineer and undertook his studies at the National Technical University of Athens. He then undertook his Masters degree in Artificial Intelligence in Edinburgh. He has publications in the area of Artificial Intelligence and Genetic Algorithms applications mainly

John Vrettaros, John Pavlopoulos, George Vouros, Athanasios Drigas

in the area of medicine and education. He is currently involved in research programs at N.C.S.R. Demokritos.

George Vouros (B.Sc, Ph.D) is currently a Professor and Dean of the School of Sciences, University of the Aegean, Greece, Director of the MSc Program on Information and Communication Technologies and President of the Hellenic Society of Artificial Intelligence. His published scientific work includes more than one hundred book chapters, journal and national and international conference papers in the above mentioned themes. He has served as program chair and chair and member of organizing committees of national and international conferences on related topics.

Athanasios Drigas is a Senior Researcher at N.C.S.R. Demokritos. He is the Coordinator of Telecoms and founder of Net Media Lab since 1996. From 1985 to 1999 he was Operational manager of the Greek Academic network. He has been the Coordinator of Several International Projects, in the fields of ICTs, and e-services (e-learning, e-psychology, e-government, e-inclusion, e-culture etc). He has published more than 200 articles, 7 books, 25 educational CD-Roms and several patents. He has been a member of several International committees for the design and coordination of Network and ICT activities and of international conferences and journals.

1 Introduction

According to recent research (Lytras, 2007; Lytras and Sicilia, 2005) many positive aspects have come up for a virtual learning community via e-learning and that is why its popularity has grown rapidly. Indeed, implementing e-learning turned out to be a fruitful way to achieve a higher knowledge level in a particular field especially when we take into account the fact that the participants form a geographically dispersed group of people.

However, in order to maximize the positive results of e-learning some kind of assessment is more than necessary. There is no doubt that the asynchronous character of e-learning is another obstacle to effective tutoring as the diagnosis of a student's cognitive abilities demands constant interaction. This is the reason why software developers try to create a system which simulates human teacher behavior and especially the way the instructor adapts to the individualization of each student (Vargas-Vera and Lytras, 2008)). Consequently, artificial intelligence is the answer to developers' expectations and several approaches have already come up. Common methods used so far are fuzzy logic techniques to diagnose a student's knowledge level and neural networks for simulating as well as monitoring a learner's cognitive process (Stathacopoulou, Magoulas, Grigoriadou and Samarakou, 2005; Vrettaros, Vouros and Drigas, 2007).

This paper presents the development of an assessment system of the gained knowledge of students. In specific, the results of self-assessment exercises provided by a learning environment are examined, in order for the students to obtain the knowledge level they have possessed in each learning section solely and overall. The final aim is for the assessment system to be trained in order to play the role of an instructor. The assessment system was based on a novel implementation of a Neural Network approach

....

optimized with the aid of Genetic Programming in order to be able to interfere and adjust the Neural Networks' architecture to the specific problem. Thus, novel solutions could be suggested in future work which could be adjusted better to the assessment problem. The final purpose is for the system to be able to operate as an E-Tutor.

NNs are weighted interconnected networks of artificial neurons (computational models based on the biological neuron). The training procedure consists of modeling the structure of the NNs as well as defining the values of their weights. Although a gradient descent algorithm such as back-propagation is most often used as a training algorithm, an evolutionary algorithm such as GP has the potential to produce optimal network architectures in such a way that they will consist of the appropriate inputs, connections and weights for a given data set (Koza and Rice, 1991; Koza, 1995; Spears, 1991; Siddique and Tokhi, 2001). Thus, training of the NNs could be completed without the trial and error procedure and hence, forward the implementation of autonomous systems. Such a hybrid methodology is GPNN, which produces an initial population of randomly generated NNs and then recombines them through GP operations (reproduction, crossover and mutation) in order for the fittest to survive. The extracted NN is considered to be the most appropriate one for the generalization of the input pattern to the output pattern.

GPNN was initially implemented by Ritchie et Al. with the Lisp programming language in order to study the genetic underlay of diseases (Ritchie et Al, 2003; Ritchie et Al, 2007). Thereafter, the methodology was re-implemented in Matlab1 in order to study both the genetic and the environmental underlay of diseases. Until now, GPNN methodology has been used as a powerful statistical pattern recognition tool in the Bioinformatics field (Ritchie et Al, 2003; Ritchie et Al, 2007). Ten, binary-node, GPNN models were extracted each time and the most frequent group of factors was considered to be related to the outcome (e.g. presence or absence of disease). In this paper however, a novel operation of the GPNN methodology is suggested. In order for the system to be able to model the work of a pedagogical expert, NNs were built so that they would result a ranking-node instead. This final node would then represent the learners' assessment.

GPNN Assessment System (GPNNAS) is a GPNN system trained with the answers of students and their evaluation according to a pedagogical expert. The data generated by the learner going through a mini-test consists of a string of characters and values which are built based on certain criteria. The types of questions are both single-select and multi-select and have several answer options. The questions test learners against more than one sector while each question has a relevance value against every sector. The final purpose is for the GPNNAS to be able to evaluate the answer of a student according to various learning areas (criteria) as well as lead to an automated self assessment system that could train itself to different kinds of tests without human supervision.

2 Data of the expert system

The data of the developed system are real data that were extracted from the answers of learners from the Dedalos1 educational project. The modeling of the data was proved to be precise.

Dedalos learners undertook a mini-test at the end of each module to assess their understanding of the learning points covered. Each mini-test comprises a series of multiple choice questions and each answer option selected provides the GPNN Assessment System (GPNNAS) with two types of data: test data and training data. Pedagogical experts have assigned educational values to the test and training data which, in turn, allows GPNNAS to assess the learner's understanding of the module. The rest of this section describes these two data types and how values are assigned to them.

2.1 Purpose and transmission of test data

Test data assesses how relevant a question is against one of the following areas of learning:

- A. letter recognition and alphabetical order
- B. spelling/vocabulary
- C. grammar/sentence structure
- D. reading
- E. writing

Test data also evaluates the answer options against the five areas of learning and specifies whether the answer is correct, partially correct or incorrect.

2.2 Assignment of test data

Firstly, each question is assigned a relevance value between 0 and 4 by a pedagogical expert. For example, the question "Which sign is in capital letters?" mainly tests the learner's skills in section A and hence, receives a relevance value of 4 here. It is also about an underpinning reading skill at a low level and therefore it is given a relevance value of 1 in section D. It does not test spelling/vocabulary, grammar/sentence structure or writing at all, hence these sections receive a relevance value of 0.

Table 1 Relevance of questions.

Section Code	Section name	Relevance Value
A	Letter recognition and alphabetical order	4
B	Spelling/vocabulary	0
C	Grammar/sentence structure	0
D	Reading	1
E	Writing	0

1 Dedalos: Teaching English as a second language to deaf people, whose first language is sign language, via e-learning tools. LEONARDO DA VINCI, Community Action Programme on Vocational Training, Second phase: 2000- 2006

....

Secondly, each answer option is assigned evaluation values. Evaluation values are also set against the five learning areas. However, the mini-tests comprise two types of multiple choice questions: single select and multi select. While the principle behind the assignment of evaluation values remains the same, a different form of the data set is sent to GPNNAS for each question type.

In single select questions there is only one correct answer. For example for the question “Which sign is in capital letters?” option 2 is the only correct answer and the evaluation values are assigned as follows:

Table 2. Evaluation of answers in single-select type questions.

Answer Code	Answer Options	Correct /Incorrect	Evaluation Values				
			A	B	C	D	E
1	Open	Incorrect	0	-1	-1	0	-1
2	NO ENTRY	Correct	1.0	-1	-1	1.0	-1
3	Closed	Incorrect	0	-1	-1	0	-1
4	Staff Only	Incorrect	0.3	-1	-1	0.3	-1

Hence:

- **1.0** is assigned to cell 2A because the answer option is correct and the question is relevant to area A - Letter recognition and alphabetical order.
- **1.0** is assigned to cell 2D because the answer option is correct and the question is relevant to area D - Reading
- **0.3** is assigned to cell 4A because the answer option ‘Staff Only’ is partially correct as it contains two capital letters and the question is relevant to area A
- **0.3** is assigned to cell 4D because the answer option ‘Staff Only’ is partially correct as it contains two capital letters and the question is relevant to area D
- **0** is assigned where an answer option was wrong but the question is relevant to the learning area
- **-1** is assigned where the question is not relevant to the learning area

In multiple select questions there can be two or more correct answers. For example, for the question “Which of these are capital letters?” there are three correct answers (options 2, 3 and 6) and the evaluation values are assigned as follows:

Table 3. Evaluation of answers in multi-select type questions.

Answer Code	Answer Options	Correct /Incorrect	Evaluation values				
			A	B	C	D	E
1	v	Incorrect	0	-1	-1	0	-1
2	G	Correct	1.0	-1	-1	1.0	-1
3	C	Correct	1.0	-1	-1	1.0	-1
4	p	Incorrect	0	-1	-1	0	-1
5	h	Incorrect	0	-1	-1	0	-1
6	B	Correct	1.0	-1	-1	1.0	-1

The question is primarily devised to test the learner's knowledge of area A - Letter recognition and alphabetical order and to a lesser extent knowledge of area D - Reading. The following values are assigned to the correct answer options (2, 3 and 6):

- Section A: **1.0** – because the answer is correct and the question is relevant to this area
- Section D: **1.0** – because the answer is correct and the question is relevant to this area
- Sections B, C and E: **-1** because the question is not relevant to these areas

3 Methodology

With the outstanding dissemination of e-learning and the participation of numerous geographically dispersed students, a number of questions came up concerning the constant monitoring of the course of each individual's learning, the evaluation of his / her further progress as well as the adjustment of the e-learning platform to the needs of the profile of every student. In order for all the above to be achieved, the e-learning platform must be equipped with a powerful assessment tool which will be able to substitute an instructor in the evaluation of the student. For this reason, systems of artificial intelligence are being applied in fuzzy logic techniques, neural networks and genetic programming.

Next are being presented some interesting applications:

Bayesian Networks have been used in order to achieve diagnostic, cognitive assessment (Zhang and Leung, 2007). Indeed, according to the findings, Bayesian Networks facilitate valid and interpretable diagnostic feedback on performance as well as the monitoring of the progress in mastering complex domains.

Furthermore, a personalized intelligent tutoring system based on the proposed fuzzy item response theory (FIRT) obtains a more accurate evaluation of every student's individual progress and an estimation of his / her comprehension percentage (Chih-Ming and Ling-Jiun, 2008). Experiment results indicate that applying the proposed FIRT to web-based learning can provide better learning services for individual learners.

Neural Networks have been used in order to develop a fuzzy logic-based model of the diagnostic process (Stathacopoulou, Magoulas, Grigoriadou and Samarakou, 2005). This model was implemented as a means of a reliable evaluation of student's comprehension. This study has successfully simulated the diagnostic process.

Finally, a developed system, which simulates the SOLO taxonomy, obtains the assessment of the "mathematical" age of a student, using fuzzy techniques (Vrettaors, Vouros and Drigas, 2007).

However, NNs could be built in a way so that they would represent the learners' assessment. Furthermore, Genetic Programming could achieve quick convergence to the solution. Thus, GPNN could become a very useful tool for the implementation of a self assessment system.

3.1 Genetic Programming Neural Networks

GPNN was initially developed by Richie et al. (2007) to improve upon the trial-and-error process of choosing an optimal architecture for a pure feed forward back propagation NN. However, the methodology was re-implemented at the Biosim Lab of

....

the National Technical University of Athens, Greece in order to study the genetic and environmental underlay of diseases. In this paper is presented an application of this implementation which aims at training a system, through an automated procedure, to evaluate learners' answers according to a number of criteria.

GPNN has adopted the use of binary expression trees in order to allow GP to evolve a tree-like structure that adheres to the components of a NN (Fig.1) (Ritchie et Al, 2003; Ritchie et Al, 2007). The GP was constrained to use standard GP operators as well as to retain the typical structure of a feed-forward NN. Furthermore, rules were defined to ensure that the GP tree would maintain the structure that represented a NN (Koza and Rice, 1991; Koza, 1995).

Figure 1 The tree structure of a Neural Network. The o-node is the output node, the w-node is the weight node, the s-node is the activation function node and the x-node is the input node which in this case is non binary.

The steps of the GPNN method are described in brief as follows. In step one, GPNN has a set of parameters that must be initialized before the beginning of the evolution of the NN models. These include, an independent variable input set, a list of mathematical functions, a fitness function, and finally the operating parameters of the GP. These operating parameters include the population size and the number of generations. In step two, the training data are modeled according to the tested problem. In step three, the training of the GPNN begins by generating an initial population of random solutions. Each solution is a binary expression tree representation of a NN (Fig.1). In step four, each GPNN is evaluated on the training set and its recorded fitness. In step five, the best solutions are selected for crossover and reproduction, using a fitness-proportionate selection technique, called roulette wheel selection, based on the classification error of the training data (Ritchie et Al, 2003; Ritchie et Al, 2007). Classification error is defined as the proportion of individuals for whom the output was incorrectly specified. A predefined proportion of the best solutions are directly copied (reproduced) into the new generation. Another proportion of the solutions are used for crossover with other best solutions and finally the last solutions are mutated. The extracted NN, which is the best-so-far solution, is considered to be capable of classifying the data with the minimum error. In the last step, the best-so-far solution is being held and the new generation, which is equal in size to the original population, begins the cycle again. This continues until some criterion is met, and at that point the GPNN stops. This criterion is either a classification error of zero (best-so-far solution) or the maximum number of generations reached (error message).

In previous applications, 10 GPNN final models were extracted and were being used to capture any patterns which were inside the training data set. This means that if a group of factors was considered to be important in the data, it should also take part to a significant number of final GPNN models. This implementation is also the reason why GPNN has been applied successfully to the Bioinformatics field (Ritchie et Al, 2003; Ritchie et Al, 2007). However, GPNN could also result one final model that it could be used for a number of different tasks, such as classification problems. This application makes use of this observation and uses the GPNN methodology as a classification algorithm.

3.2 Application of GPNN

Until now, GPNN was mostly used for pattern recognition in the field of Bioinformatics (Ritchie et Al, 2003; Ritchie et Al, 2007). However, this GPNN application aims at modeling the classification of the answers of learners and thus, each Network is required to give the same (or approximately the same) score for a set of answers as the human scoring function. In order to model effectively the evaluation of the questions and allow for generalization over various data sets (e.g. consistent and inconsistent such as datasets which lead to the same input but different evaluation), one NN was trained for each criterion of a question. This technique ensures consistency over the evaluation while the computational cost remains linear over the input, as far as the assessment procedure (after the training of the NNs) is concerned.

The training procedure of the assessment system for each question consisted of training six NNs, one for each of the five criteria and one for the overall performance.

Figure 2 The steps of the Assessment System

The inputs of the NNs (answer patterns) consisted of binary strings representing different answer codes. Inside the binary string, the 1's represented the correct choices of the learner while the 0's the wrong ones. For example, the NN input string 1-0-0-0, for a single select question, would indicate that the learner selected the first choice as the correct one. The output of each NN (answer evaluation) could either be negative, indicating an irrelevant criterion, or a number from the space [0,1], representing the evaluation of the learner's answer according to the specific criterion.

GPNNAS, in its pattern operation has been applied for both a question of single select and a question of multi select type and has modeled the data successfully proving the system's capability of modeling this kind of data. The single select type question was "Which sign is in capital letters?" and there were four possible answers, while the multi select type question was "Which of these letters are capital letters?" and there were nine possible answers.

Table 4. The question "Which of these letters are capital letters?" and the encoded array that was used to train the system

Which of these letters are capital letters?															
Test id	Question id	Type	Relevance:	A	B	C	D	E							
t1	q1	multiple select		4	0	0	1	0	correct answer						
									011001						
Answer option	Answer option id	Correct?	Evaluation:	A	B	C	D	E	user answer	Training Data					
v	1	FALSE		0	-1	-1	0	-1	011001	1	-1	-1	1	-1	1
G	2	TRUE		1	-1	-1	1	-1	111000	0	-1	-1	0	-1	0
C	3	TRUE		1	-1	-1	1	-1	101001	0	-1	-1	0	-1	0
p	4	FALSE		0	-1	-1	0	-1	110001	0	-1	-1	0	-1	0
h	5	FALSE		0	-1	-1	0	-1	any other	0	-1	-1	0	-1	0
B	6	TRUE		1	-1	-1	1	-1							

....

For the single select type questions, the initial NNs population was set to be 10 NNs while for the multiple select type questions 100 NNs. Furthermore, the generations of the Genetic Programming evolution were set to be 50. During the training procedure, a plot function depicted the training procedure for each criterion. The training procedures for the first three criteria of a single select type and a multiple select type question are depicted in Fig.3 and Fig.4 correspondingly.

The list given below represents the Neural Network with the use of Matlab:

ans =

Columns 1 through 12

```
3.0000 4.0000 4.0000 1.6000 3.0000 2.0000 3.0000 0 0 4.0000
4.0000 1.6000
```

Columns 13 through 24

```
7.0000 4.0000 4.0000 0 0 0 0 5.0000 2.0000 5.0000
7.0000 0
```

Columns 25 through 36

```
0 1.7000 1.8000 5.0000 7.0000 5.0000 2.0000 0 0 0
0 0
```

Columns 37 through 48

```
0 0 0 102.0000 0 1.9000 2.0000 104.0000 0 7.0000
7.0000 0
```

Columns 49 through 60

```
0 0 0 0 0 0 0 103.0000 0 1.7000 6.0000
101.0000
```

Columns 61 through 72

John Vrettaras, John Pavlopoulos, George Vouros, Athanasios Drigas

0 0.7000 1.6000 0 0 0 0 0 0 0 0 0

Columns 73 through 84

0 0 0 0 0 0 0 0 0 0 0 0

Columns 85 through 96

0 2.0000 1.3000 0 0 0 0 7.0000 1.3000 8.0000
0.6000 0

Columns 97 through 108

0 0 0 0 0 0 0 0 0 0 0 0

etc.

Figure 3 Training procedure for a single select type question

Figure 4 Training procedure for a multiple select type question

The answers of the learners were uploaded via a web page to the main server (Fig. 5), wherein they were encoded in an appropriate form and were processed by the GPNNAS.

Figure 5 The question interface for the Dedalos e-learning environment

The output of the system was the learner's evaluation for the five criteria examined as well as for the learner's overall performance. Furthermore, the evaluation was presented to the learner through a bar diagram (Fig. 6), forwarding intelligibility of the results for the user.

Figure 6 Classification form of the results

4 Discussion

This study examined the application of a Neural Network Genetic Programming approach over the self-assessment procedure of learners. The final purpose was to pose the ground for a fully automated intelligent self assessment system that could model the evaluation of learners according to experts.

The data used in this study did not allow for an extensive research of the power of such a system, as far as computational intelligence is concerned. This means that the generalization of the system was not researched deep enough. However, this was not the major goal in this study and is not a drawback of the application since the generalization abilities of GPNN are widely studied and recorded in even larger and more complex input spaces (Ritchie et Al, 2003; Ritchie et Al, 2007). Instead, this paper focused to study the abilities of such a self assessment system when an automatic procedure is incorporated. A

....

more extensive study of the system's abilities then, as far as its power is concerned (e.g. system's lower boundaries on generalization over data), could be a different problem.

The system should be able to cope with various different data sets automatically and simulate the expert's evaluation behavior at the same time. Moreover, in order to simulate best the expert's 'ways' of evaluation, training data should be consistent (a hard problem when various different data sets are supposed to be used). This means that two similar inputs, as for two same training inputs, should not result to different output (very common with multi select and single select problems). However, in this paper a different technique was adopted to deal with this problem according to which, one Neural Network was supposed for each evaluation criterion. Thus, a simpler task was assigned to each NN during training as well as a more consistent approach was built as far as the automatic procedure is concerned (two different criteria correspond to two different networks). This is because the generalization has been bounded to the criteria and not in the whole questionnaire.

It should also be noted that there is a reverse analogous relationship between this ability to cope with various sets and the computational intelligence of the system. However, if one would want to model more accurately the expert's evaluation techniques, one Neural Network could be trained over all criteria of a question or over all questions of the questionnaire if an even more consistent scheme is preferred. This could be done if consistency of the data was verified first but even so, this would result to less different data sets which could be presented to the system (else, same inputs will be dealt with fuzzy methods since the NNs will not be trained to deal with these data) .

In order to implement such a system, a more exhaustive technique, such as Back Propagation algorithm, could be applied to Neural Networks instead of Genetic Programming. Then, a possible over fitting of the data, could in this case, lead to better results. This is because a better simulation of the expert's mind could be achieved, as far as the specific questionnaire is concerned. Then, a dual task system could be implemented allowing the user to choose the way of training his data and leading to a more functional e-learning system.

5 Conclusions

In this paper, a hybrid expert system with use of GPNN is developed for the evaluation of learners' answers according to a number of criteria. Thus, the assessment data could be represented to the learner in a meaningful and useful way in order to help the learner improve his skills in the cognitive sections where he showed low performance in the relevant test. The application of the GPNN methodology for e-learning purposes allows for generalization of the assessment process which could lead to the implementation of an intelligent e-Tutor. The system was applied and evaluated successfully learners' answers, which were derived from an educational project for the teaching of English as a second language to deaf people whose first language is the sign language. The next challenge is a fully automated training procedure wherein the training data will be presented to the assessment system online and the system could be trained in real time, as well as over different and more complicated kinds of tests. Thus, an e-learning system could be implemented which could serve various kinds of learners who need to improve their learning abilities according to various criteria.

John Vrettaros, John Pavlopoulos, George Vouros, Athanasios Drigas

References

- Chih-Ming C., Ling-Jiun D. (2008) 'Personalized web-based tutoring system based on fuzzy item response theory', *Expert Systems with Applications: An International Journal (Elsevier)*, Vol. 34, No. 4, pp.2298-2315.
- Koza J. R. (1995) 'Survey of genetic algorithms and genetic programming', In: *WESCON/95 Conference Record: Microelectronics Communications Technology Producing Quality Products Mobile and Portable Power Emerging Technologies*, pp. 589-594.
- Koza J. R., Rice J. P. (1991) 'Genetic generation of both the weights and architecture for a neural network', In: *IJCNN-91-Seattle International Joint Conference on Neural Networks*, Vol. 2, pp. 397-404 .
- Lytras, M.D. (2007) 'Teaching in the knowledge society: An art of passion', *International Journal of Teaching and Case Studies*, Vol. 1, Nos. 1/2, pp.1-9.
- Lytras, M.D., Sicilia, M.A. (2005) 'The knowledge society: A manifesto for knowledge and learning', *International Journal of Knowledge and Learning*, Vol. 1, Nos. 1/2, pp.1-11.
- Ritchie M. D., Motsinger A. A., Bush W. S., Coffey C. S., Moore J. H. (2007) 'Genetic programming neural networks: A powerful bioinformatics tool for human genetics', *Applied Soft Computing (Elsevier)*, Vol. 7, No. 1, pp. 471-479.
- Ritchie M. D., White B. C., Parker J. S., Hahn L. W., Moore J. H. (2003) 'Optimization of neural network architecture using genetic programming improves detection and modeling of gene-gene interactions in studies of human diseases', *BMC Bioinformatics*, Vol.4, No. 1, rec.NO 28.
- Siddique, M. N. H. and Tokhi, M. O. (2001) 'Training Neural Networks: Backpropagation vs. Genetic Algorithms', In: *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN '01)*, Vol. 4, pp. 2673-2678.
- Spears W. M. (1991) 'A Study of Crossover Operators in Genetic Programming', In: *Proceedings of the 6th International Symposium on Methodologies for Intelligent Systems (Lecture Notes In Computer Science)*, Vol. 542, pp. 409-418.
- Stathacopoulou, R., Magoulas, G.D., Grigoriadou, M., Samarakou, M. (2005) 'Neuro-fuzzy knowledge processing in intelligent learning environments for improved student diagnosis', *Information Sciences-Informatics and Computer Science: An International Journal (Elsevier)*, Vol. 170, No.2-4, pp. 273-307.
- Vargas-Vera, M., Lytras, M.D. (2008) 'Exploiting semantic web and ontologies for personalized learning services: towards semantic web-enabled learning portals for real learning experiences' *International Journal of Knowledge and Learning*, Vol. 4, No.1 pp. 1-17.
- Vrettaros, J., Vouros, G., Drigas, A. (2007) 'Development of an Intelligent Assessment System for Solo Taxonomies Using Fuzzy Logic', *Symbolic and Quantitative Approaches to Reasoning with Uncertainty (Lecture Notes in Computer Science)*, Vol. 4724/2007, pp. 901-911.

....

Zhang, Z. & Leung, K. (2007) 'Design of Diagnostic Cognitive Assessment for Web-Based Learning Environments with Bayesian Network Models', *In: Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, pp. 6701-6706.

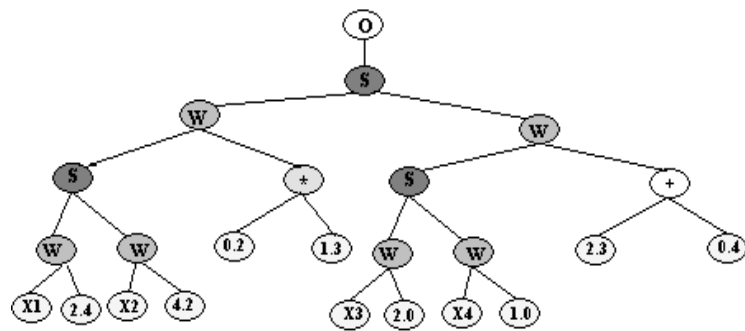


Fig.1 The tree structure of a Neural Network. The o-node is the output node, the w-node is the weight node, the s-node is the activation function node and the x-node is the input node which in this case is non binary.

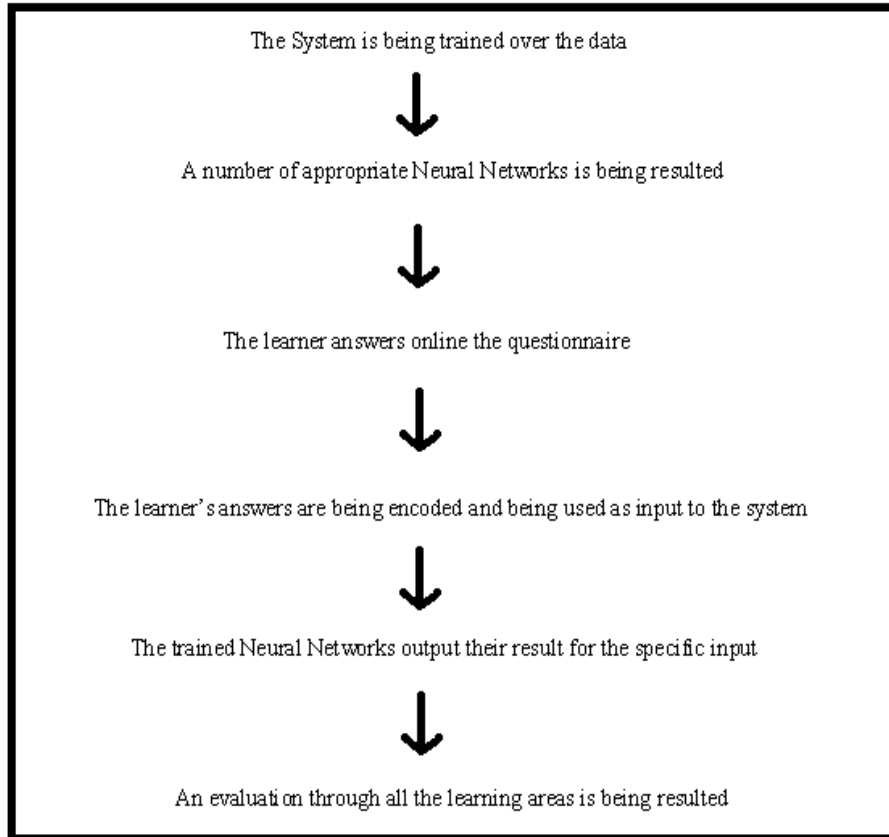


Fig. 2 The steps of the Assessment System

....

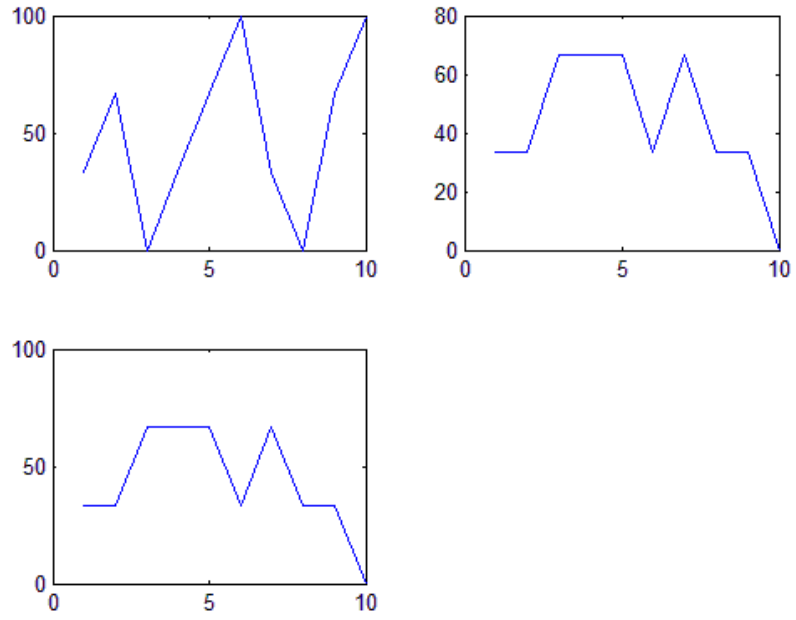


Fig. 3 Training procedure for a single select type question

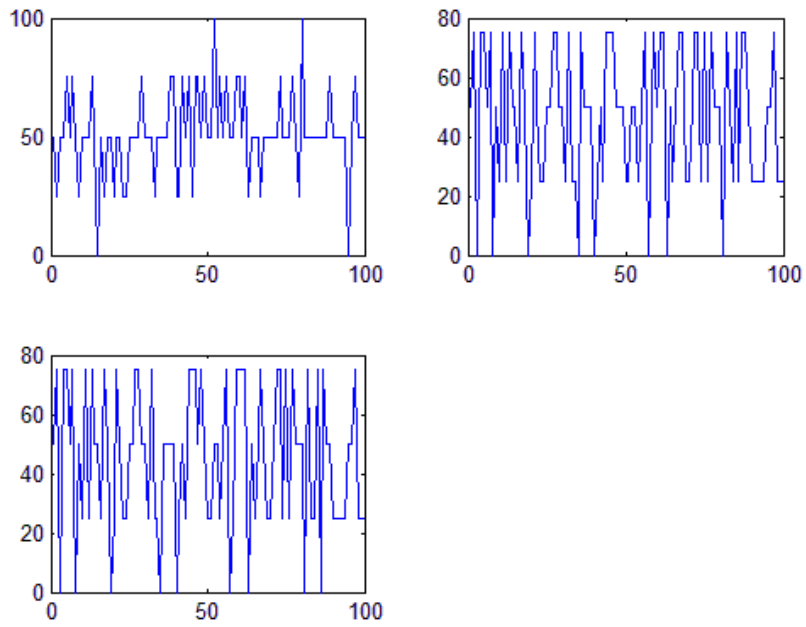


Fig. 4 Training procedure for a multiple select type question

John Vrettaros, John Pavlopoulos, George Vouros, Athanasios Drigas

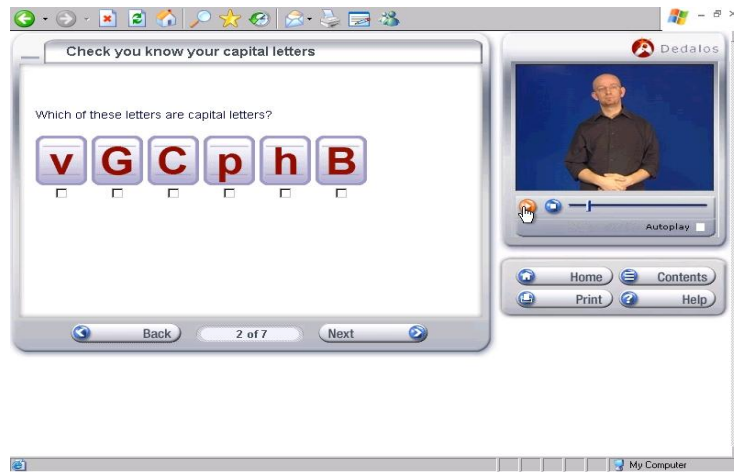


Fig. 5 The question interface for the Dedalos e-learning environment

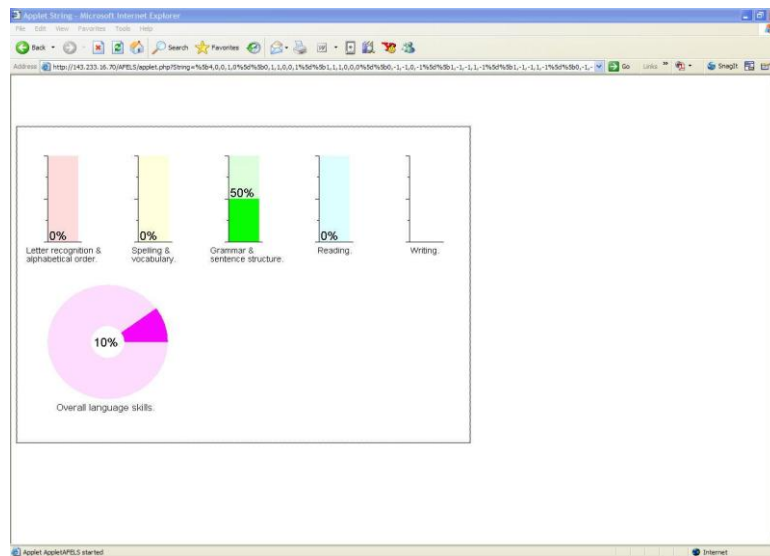


Fig. 6 Classification form of the results