An Improved Pattern Classifier Based Thyroid Disease Prediction System

**Abstract:** *A challenging task in the medical science is to attain the accurate diagnosis of disease prior to its treatment. Pattern classifiers are used for solving complex and non separable problems in different fields like biochemical analysis, image processing, chemical analysis etc . In this paper , a multilayer perceptron pattern classifier is proposed for the diagnosis of thyroid disease. The whole proposed framework is implemented in MATLAB with real data of thyroid patients.*

**Keywords:** *Software engineering education, reflective learning and teaching.*

1. **Introduction**

An Artificial neural network is an information processing paradigm that is motivated by the way natural neural system i.e. brain process data. The neural network constitutes of countless interconnected information handling components called neurons. The key component of the neural network is a novel structure. Neural systems, with their noteworthy capability to derive meaningful information from imprecise information, can be utilized to separate and distinguish patterns that are too intricate to be noticed by any computer technique or by humans.

**Input layer**

**Hidden layer**

**Output layer**

**I1**

**I2**

**In**

**O1**

**Om**

 Figure 1. Architecture of Neural Network

As NN is a self learning framework, it show distinctive classes of learning calculations, for example, Supervised learning, Unsupervised learning and Reinforcement learning.

**MACHINE LEARNING**

**UNSUPERVISED LEARNING**

**SUPERVISED LEARNING**

**CLASSIFICATION**

**REGRESSION**

**CLUSTERING**

 Figure 2.

*1)Supervised learning*

In this algorithm calculated desired output for the system is additionally given to the system while the system is being trained. By giving the neural system both an input and desired match, the error can be figured out by ascertaining the difference between the objective yield and genuine yield. It would then be able to utilize that error to make changes to the system by updating it's weights.

*2) Unsupervised Learning*

In this algorithm a set of inputs is given to the neural network and it is the responsibility of neural network to recognize some kind of pattern present in the inputs . This sort of learning is used worldwide as a part of information mining.

*3) Reinforcement Learning*

This learning algorithm is similar to supervised learning in the manner that a feedback is given back ,and at the end on the basis of how well the system performed ,a reward is given. NNs are widely used in the real - world computation applications. The various areas of application include pattern recognition, pattern classification and pattern prediction. The whole paradigm of predicting lifestyle disease is shifting from conventional method to computer based expert prediction system.

 Thyroid disease is one among the common lifestyle diseases. Thyroid organ is a butterfly-molded organ which is present in the neck underneath the mouth of human body. It release hormones that control metabolism like heart rate, body temperature etc.It produces two main hormones T3 and T4*.*These hormones are responsible for various metabolic activities like body weight, heart rate etc. These activities may get disturbed if the level of these hormones change. So the diagnosis of thyroid disease is important prior to its treatment. About 32 percent of the total Indian population suffer from thyroid disease. Thyroid disease can be broadly categorized i.e. hypothyroid and hyperthyroid.

THYROID

**NORMAL**

**HYPOTHYROID**

**HYPERTHYROID**

 Figure 3.

 When the amount of hormones exceed the amount required by the human body, it causes hyperthyroidism. Hypothyroidism is the inverse of hyperthyroidism; it reduces body metabolism, cause drowsiness and pain in joints.



 Figure 4.

According to a survey, almost every Indian suffer from one or the other type of thyroid disease case which results into hormonal imbalance and gain in weight. As shown in the figure 3, a woman namely Noorjahan was suffering from severe hyperthyroid disease. On 17th July 2015 she was operated in which her thyroid gland was removed which weighed about 3 kg that time. In the early stage of hyperthyroidism, she faced no problem but with the time her neck got deformed and the enlarged thyroid organ was close to the two pipes as shown in the picture ,putting pressure on critical veins and blood vessels.



 Figure 3.Woman suffering from severe hyperthyroid disease

**2.Background**

Various scholars have used different pattern classifiers for developing lifestyle disease prediction systems. In this section a brief study of thyroid disease prediction system have been presented.

I.MD.Dendi et.al.(2015) have used six different methods for the diagnosis of thyroid disease. After experimenting, the performance of multilayer perception was found highest as compared to that

of the other five methods.

Mohd. Reza et.al. (2017) have discussed the diagnosis of different types of thyroid disease using neural networks by considering their age. The inputs for the thyroid prediction system are seven hormone tests including age and the output is the diagnosis of the thyroid. The various network structure used include MLP, PNN, GRNN AND CFNN.

Shivanee panday et al(2016) have proposed various data mining techniques like Bayes net, multilayer perception, RBF network, L4.5, CART, REP tree, decision stump to develop classifiers for diagnosis of hypothyroid disease. After performing experiments, it is clear that REP tree and L4.5 techniques perform well as compared to others.

Mazin Abdul rasool hameed et.al(2017) have proposed a method of classifying thyroid disease using multilayer feed forward using back propagation learning rule .In this work three inputs have been considered as T3,T4,TSH.

Saeed Shariati and Mahdi Motanali Haghighi (2010) have used fuzzy system to diagnosis hepatitis and thyroid disease. The results of fuzzy neural networks with support vector machine and artificial neural network were compared.

Anupam Shukla et.al. (2009) in their work have trained the system using three ANN algorithms, the backpropogation (BPA), the radial basis function (RBF) and the learning vector quantization (LVQ) systems . The networks are compared on the basis of the factors like accuracy of network and training time. On comparison of the performance LVQ network is found to be have the best accuracy with 98% but among the three, RBFN when trained on hybrid dataset have the least training time.

Table 1. Summary of pattern classifiers for thyroid disease prediction.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|

|  |
| --- |
|  |
| **Authors**  |

 |

|  |
| --- |
|  |

**Year** |

|  |
| --- |
|  |

**Problem** |

|  |
| --- |
|  |

**Network used** | **Performance parameters**

|  |
| --- |
|  |

 |
| I.M.D,Hanung Adi,Noor Ahmad | 2015 |

|  |
| --- |
| Diagnosis of thyroid  |

 | MLPBack propogation |

|  |
| --- |
| Accuracy =MLP (96.7%)  |

 |
|

|  |
| --- |
| Mohd.Reza,Ali ,Omid |

 | 2017 | Diagnosis of thyroid |

|  |
| --- |
| MLPPNNGRNNFTDNNCFNN |

 |

|  |
| --- |
| Accuracy=GRNN (99.5%)  |

 |
|  Shivanee , Rohit  S R Tandan | 2016 | Parkinson's disease Diagnosis  | Bays net ,MLP,RBF,C4.5,CARTDecision tree | Accuracy=REP(99.57%)C4.5 (99.57%),Bays net (98.59%) |
|  Mazin Abdul Rasool | 2017 | Diagnosis of thyroid | MLP |  **--** |
| Saeed shariati, Mahdi motavali  |   2010 | Diagnosis of hepatitis and thyroid | Self organized fuzzy system |  **--** |
| Tiwari, RituShukla, AnupamKaur, Prabhdeep | 2009 | Diagnosis of thyroid | Backpropogation NNRBF NNLVQ NN | Accuracy =LVQ (98%) |

**3.Problem Formulation**

In order to address the major research gaps, the need is to design an improved thyroid disease pattern classifier system by including more parameters like age group, heredity, antibodies. The system may utilize better classifiers in order to improve the overall accuracy of the diagnostic system.

**4.Proposed System**

 **4.1 Framework for the proposed system**

In this section the detailed framework and algorithms are presented. The training and testing phases of the thyroid disease prediction system is clearly shown in the Figure 5.

 As shown in the figure, the first step is to identify the typical parameters / risk factors which are responsible for the thyroid disease in human beings. In the next step , miscellaneous dataset of various patients of different categories is collected. In the conventional methods of thyroid diagnosis system, majority of authors have used only three factors namely T3,T4 and TSH. In the proposed diagnostic prediction system ,more number of risk factors can be included.

 Choose Random Samples Patient

 Dataset

 Thyroid Patient

 Dataset

Choose Disease Attributes

Choose Disease Attributes

Preprocessing Of Data

Preprocessing Of Data

Choose the Pattern Classifier

Choose the Pattern Classifier

Testing The Pattern Classifier

Train The Pattern Classifier

Compute The Confusion Matrix

Figure 5. Framework for the proposed system

 In order to classify a particular patient into any of the three classes a dataset of 120 samples has been prepared. In order to remove anomalies, noise and to quantify Boolean values the data set is manually enriched.

 Once the dataset is prepared, a multilayer pattern classifier model is created and trained with the dataset. The MLP pattern classifier model is stored for the testing phase. In order to check the accuracy of thyroid predictions a sample of randomly chosen patients is applied on the stored MLP prediction system.

 **4.2** **Proposed algorithm**

 MLP is one of the most common ANN which is widely used for different tasks like pattern classification, pattern recognition etc. One of the most important feature of MLP is that we can specify any number of output classes. The network architecture chosen for this problem is MLP having eleven input nodes and three output nodes. Each node present in the input layer is connected to every other node in the hidden layer through some weights. The value of the weighted input sum to a particular node maybe large, therefore it is important to scale down the weighted sum by reducing it before producing the resulted output of that particular node. For this purpose a function is applied on the weighted input .One of the best method is backpropogation method which works on the principle of gradient descent rule.

**4.3** **Pseudo code**

**4.3.1** The pseudo code for the training the proposed algorithm is as under:

1. Initialize weights and learning rate.

2. Perform steps 3 to 10 till condition is false.

3.Repeat steps 4 to 9 for each pair to be trained.

4.Each input node receives an input signal say Xi and pass it to the next node present in hidden layer.

5.Each node in the hidden layer say hj sums its weighted input to calculate net input as(feed forward phase I):

$$Zinj=voj+\sum\_{i=1}^{n}Xivij (1)$$

Activation function is then applied on the $zinj$ to calculate the output of the hidden node:

 $Zj$ $=$ $ƒ\left(Zinj\right) (2)$

this output signal is then send as input to the output layer node from hidden node

6.For each output node Ok, calculate the total input as :

$$Oink=W0k+\sum\_{j=1}^{p}ZjWjk (3)$$

Now ,apply the activation function on $Oink$ to compute the output signal

 $Ok$ $= ƒ\left(Oink\right) (4)$

*Back-propogation learning rule (Phase II):*

7.Each output node receives the input training vector associated with the target pattern and compute the error:

 $∆ k=\left(tk-Ok\right)f^{'}\left(Oink\right) (5)$

On the basis of error calculated, adjust the weights as :

 $ ∆ Wjk=α ∆k Zj (6)$

Send $ ∆k $ back to the hidden layer

8.Each hidden node calculate the sum of this delta from the output node

 $ ∆inj=\sum\_{i=1}^{m}∆ kWjk (7)$

 Error is calculated as :

 $∆ j=∆injf^{'}\left(Zinj\right) (8)$

 *Adjust weight and bias as(Phase III):*

9.Each output and hidden nodes update bias and weights as :

 $Wjk\left(new\right)=Wjk\left(old\right)+∆Wjk (9)$

 $ W0k\left(new\right)=W0k\left(old\right)+∆W0k (10)$

10.Check whether the actual output equals the target output (stopping condition)

**4.3.2** *Testing procedure for the pattern classifier*

1.Repeat steps 2 to 4 for each input

2.Set the activation of input unit for Xi.

3.At hidden node say X, calculate the net input and output as :

$$Zinj=voj+\sum\_{i=1}^{n}Xivij (1)$$

 $Zi$ $=$ $ƒ\left(Zinj\right) (2)$

4.At output node, compute the output as:

$$Oink=W0k+\sum\_{j=1}^{p}ZjWjk (3)$$

 $Ok$ $= ƒ(Oink)$

**5.Experimental Results**

The experiment are conducted on the real dataset of 120 instances collected from SKIIMS Soura, Srinagar. the subjects were chosen carefully covering wide range of population including men, women, old and youngsters. The values for eleven attributes were collected for all the 120 instances and some attributes has been quantified The preprocessing of the dataset has been done in order to remove ambiguities, anomalies and errors. The pre-processed dataset is used to train the pattern classifier model using the back error propogation algorithm for the multi-layer perceptron.



Figure6.Training performance with 30 instances and 7 attributes

The above graph shows the best performance of MLP at epochs 9 on 30 instances with 7 attributes.



Figure 7.Gradient error with 30 instances and 7 attributes.

In figure8, the gradient error graph has been shown on 30 instances and 7 attributes.



Figure 8. Confusion matrix with 30 instances and 7 attributes.

In the figure shown above the confusion matrix clearly reveals the performance of the pattern classifier(MLP). The green cells in the confusion matrix represent correctly classified instances while as

the red cells represent incorrect classification. The blue box represent percentage of both correct as well as incorrect classification classes



Figure 9.ROC with 30 instances and 7 attributes.

In the above graph the receiver operating characteristic of training, validation, testing and overall accuracy of the MLP is shown with 30 instances and 7 attributes.

Figure10. Training performance with 120 instances and 7 attributes

The above graph represents the best performance of MLP at epochs 15 with an increase in the number of instances.

 

Figure11. Gradient error with 120 instances and 7 attributes.

In figure11, the gradient error as calculated by the MLP has been shown on 120 instances and 7 attributes.



Figure12. Confusion matrix with 120 instances and 7 attributes.

 Figure 12 clearly shows an increase in the training, validation, testing and overall accuracy of the thyroid prediction system. It means that the system is able to classify the patients with 99.2% of overall accuracy .



Figure 13.ROC with 120 instances and 7 attributes.



Figure 14. Training performance with 30 instances and 11 attributes.

The best performance achieved by the MLP with 30 instances and 11 attributes is shown in the figure above. The above figure clearly reveals that with increase in the number of attributes ,the performance of the diagnostic system also increases.



Figure 15. Gradient error with 30 instances and 11 attributes.



Figure 16. Confusion matrix with 30 instances and 11 attributes.

The confusion matrix of 30 instances with 11 attributes is shown in figure above. The training, validation, testing and overall accuracy is 85%,20%,40%,66.7% respectively



Figure 17. ROC with 30 instances and 11 attributes



Figure18.Training performance with 120 instances and 11 attributes



Figure19.Gradient descent with 120 instances and 11 attributes.



Figure20.Confusion matrix with 120 instances and 11 attributes.

As it is clear from the above confusion matrix that with increase in the number of attributes the overall accuracy has also been increased to 100%.

Figure21.ROC with 120 instances and 11 attributes

The proposed model of the diagnostic system has been evaluated at various numbers of training instances as shown in the Table1.In the initial step the training of the system has been carried out with 30 instances which used only 9 iterations with gradient error of 0.0314. The Table 2 clearly reveals that the training of the system with 30 samples results in an accuracy of 85% with testing accuracy of 40% and overall poor accuracy of 66.7%. The training of the proposed system with 7 attributes and 30 samples exhibits 14 iterations with gradient error of 9.95e-07. Furthermore, the number of instances were successfully increased for the training of the proposed system and performance is evaluated at 60,90 and 120 instances which is shown in the Table 1 and Table 2.

Table1: Performance of the proposed system with eleven attributes(epochs ,gradient errors)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No .of attributes | Training Attempt | No. of TrainingSets | Epochs | Network Trained | Gradient Error |
| 11 | 1 | 30 | 9 | YES | 0.0314 |
| 2 | 60 | 25 | YES | 0.00029 |
| 3 | 90 | 34 | YES | 0.000591 |
| 4 | 120 | 56 | YES | 9.23e-07 |

Table2 : Performance of the proposed system with eleven attributes(training, validation, testing, accuracy)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No. of attributes | Training Attempt | No. of Data Sets | Performance | TrainingAccuracy | ValidationAccuracy | Testing Accuracy | Overall Accuracy |
| 11 | 1 | 30 | 0.031 | 85% | 20% | 40% | 66.7% |
| 2 | 60 | 9.87e-05 | 100% | 77% | 100% | 96.7% |
| 3 | 90 | 0.000158 | 100% | 100% |  100% |  100% |
| 4 | 120 | 3.10e-07 | 100% | 100% | 100% | 100% |

Table3:Performance of the proposed system with seven attributes(training, validation, testing, accuracy)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No.of attributes | Training Attempt | No .of Data **Sets** | Performance | Training**Accuracy** | Validation**Accuracy** | TestingAccuracy | **Overall Accuracy** |
| 7 | 1 | 30 | 0.00209 | 75% | 100% | 20% | 70% |
| 2 | 60 | 0.0124 | 95.2% | 100% | 77.8% | 93.3%s |
| 3 | 90 | 0.000663 | 100% | 100% | 92.9% | 98.9% |
| 4 | 120 | 0.00852 | 100% | 94.4% | 100% | 99.2% |

Table4: Performance of the proposed system with eleven attributes(epochs ,gradient errors)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No.of attributes | TrainingAttempt | No. of Data**Sets** | **Epochs** | **Network Trained** | **Gradient** **Error** |
|  7 | 1 | 30 | 14 | yes | 9.95e-07 |
| 2 | 60 | 13 | yes | 0.0118 |
| 3 | 90 | 21 | yes | 0.00161 |
| 4 | 120 | 21 | yes | 0.0103 |

The Table 3 shows the training performance of the proposed system with 7 attributes at different number of instances. The overall accuracy of the proposed diagnostic system is 100% where as 99.2% with seven attributes. On the other hand, the results clearly reveals that the number of attributes for thyroid diagnosis are independent of the number of instances in terms of overall accuracy which is 100% in both the cases. Surprisingly, with lower number of instances for training the model (e.g. 30), the overall accuracy is better for seven attributes(70%) instead of eleven attributes(66.7%). The number of epochs required

for training the model with eleven attributes increases with increment in the samples.

* 1. **Conclusion**

The ultimate objective of this study was to use the results and conclusions obtained from group projects in a reflective manner, in order to improve learning and teaching of software engineering in large groups, and in particular at UWE. This study has led me to devise more controls on the management of group projects. For example, students were asked to submit an initial project proposal four weeks after the assignment was released, with emphasis on initial understanding of the problem domain, team’s structure, and initial project plan. In addition, tighter links between lectures and both practical sessions and group-work were implemented.

In general, I wish to have a tighter policy with respect to attendance in practical sessions (at least) as this had impacted group-work and attainment in practical sessions. Records of students’ attendance were kept, but because there is no link between sessions attended and passing/failing the module, it makes controlling students attendance a tall objective to achieve.

The use of debates as a mechanism by which the skills of creative and critical thinking are developed is of paramount importance to the individual, society, and employers. Nevertheless, the great returns of this approach could be hampered by the high absence of students in practical sessions given the nature of working in a team and the activities involved in software engineering.

In addition, tutors and IT support staff need training on the use of tools. Also, tutors need extra time and budget to attend seminars presented by tool vendors in order to update themselves with new features provided by the tools for later enhancements to students’ projects.

Team work and the development of team skills are vital not only for the accomplishment of the group coursework of the software engineering module but also for preparing software engineers who are ready to work within teams as it is the case in the industry. In addition, this will cultivate team spirits in individuals. Not only will this have impact on the individual, but also the society as a whole. Hence, this facilitates the achievement of one of the key objectives of educational research.

This research may be considered as generic model for other modules where attainment in exams, coursework, and feedback from lecturers, tutors, and students are studied in a reflective manner to improve on learning and teaching of a particular subject. Thus, this research could be classified as a generic one from which one can instantiate to create instances of reflective models to improve teaching and learning of different subjects in particular computing and other subjects in general.

Finally, I believe that this study is a step forward towards educational research that critically improves educational action as per Griffiths’s definition “Educational research aims critically to inform educational judgements and decisions in order to improve educational action.” [5]. In addition, the outcomes of the first loop of this reflective spiral process in 2000/2001 will evaluated further and the outcomes will be put forward for implementation in 2001/2002 in a reflective manner.

**References**

1. Boehm B., “A spiral model of software development and enhancement,” *IEEE Computers*, vol. 21, no. 5, pp. 61-72, 1988.
2. Booch G., Rumbaugh J., and Jacobson I., *The Unified Modeling Language*, Addison-Wesley, 1999.
3. Cutts Q., “Fostering engagement, covering content: reflecting on large, mixed ability, first year computer programming module,” in Walker M. (Ed), *Reconstructing Professionalism in University Teaching*, Open University Press, pp. 168-189, 2001.
4. Favela J. and Pena-Mora F., “An Experience in Collaborative Software Engineering Education,” *IEEE Software*, pp. 47-52, March/April 2001.
5. Griffiths M., *Educational Research for Social Justice*, Open University Press, pp. 38-39, 1998.
6. Kermmis S., “Action Research,” in Hammersley M. (Ed), *Educational Research – Current Issues*, Open University Press, vol. 1, 1993.
7. Krutchen P., *The Rational Unified Process Model: An Introduction*, 2nd ed, Addison-Wesley, 2000.
8. Krutchen P., “The 4+1 View Model of Architecture,” *IEEE Software,* vol. 12, no. 6, 1995.
9. Kolb D. A., *Experiential Learning: Experience as the source of learning and development*, Prentice-Hall, New Jeresy, 1984.
10. Newble D., and Cannon R., *A Handbook for Teachers in Universities and Colleges,* Kogan Page, 1991.
11. Ramsden P., *Learning to Teach in Higher Education*, Routledge, London , 1992.
12. Sommerville I., *Software Engineering*, 6th ed, Addison-Wesley, 2001.
13. Warhurst C., “Developing Students’ Critical Thinking: the Use of Debates,” in Walker M. (Ed), *Reconstructing Professionalism in University Teaching*, Open University Press, 2001.