Hetero Associative Memory Based Neural Network Classifier for Health Care Data Diagnosis

R. Akila, M. Nandhini, S N Sivanandam

Abstract— Classification is one of the predictive data mining tasks used to discover a model from the past data to predict some response of interest. In this work, Hetero Associative Memory based Neural Network (HAMNN) classifier is employed for health care data diagnosis. Classifier performance is enhanced by using Lern matrix, a popular model for associative memory. HAMNN classifier is built efficiently to improve the classification accuracy. This classifier provides promising results when experiments were conducted using six health care datasets from UCI machine learning repository.

Index Terms— Neural network, Associative memory, Hetero associative memory, Lern matrix.

I. INTRODUCTION

Diagnosis of disease is a significant task in the medical field. Disease diagnosis can be predicted manually but sometimes this may lead to some false presumptions. To conquer this difficulty, data mining techniques such as classification, rule mining and clustering technique can be employed. Data mining is the process of extracting unknown knowledge from huge volume of database. The main intention of data mining is to provide useful information that can be analyzed and processed in different ways in future. Classification is one of the important techniques which are used to predict the class of the tuple based on the model built on the training data. In literature, several classifiers such as Naïve Bayes, ID3, SVM, etc., are existing to classify the health care datasets. Though these techniques have very minimal training time, but often results in poor accuracy. In order to overcome this limitation neural network based classification is proposed for disease diagnosis. This paper mainly concentrates in implementing neural network using associative memory. Associative memories are also known as the content addressable memory which is used to map the input pattern to the output pattern. The main advantage of associative memory is that it can recall the stored pattern exactly. Auto associative and Hetero associative are the two types of associative memories. Auto Associative Memory (AAM) retrieves the stored pattern that strongly resembles the current pattern. Hetero Associative Memory (HAM) retrieves output pattern from the input pattern which is different in type, content and format. Lern matrix is one of the associative memory concept used in this paper to build the HAMNN classifier for better classification results.

The rest of the paper is organized as follows. Section 2 discusses the literature survey. Section 3 describes the methodology followed in the proposed work. Experimental results are discussed in section 4 and section 5 concludes the paper.

Manuscript Received on May 14, 2015.

R. Akila, Computer Science and Engineering, PSG College of Technology Coimbatore.

Asst.Prof M. Nandhini, Computer Science and Engineering, PSG College of Technology, Coimbatore.

Dr. S N Sivanandam, Computer Science and Engineering, Karpagam College of Engineering.Coimbatore.

II. BACKGROUND AND RELATED WORK

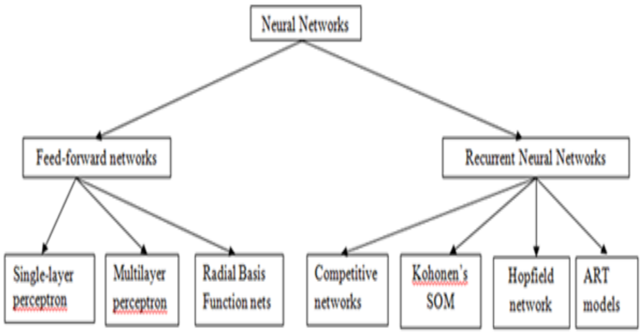

Fig. 1 Neural Network

The working of Artificial Neural Network (ANN) resembles the functioning of the human brain. ANN is the abstract computer model of the human brain. There are two types of neural network such as Feedforward and Recurrent Neural Network. Associative memories can be built either using Feedforward or Recurrent Neural Network. The main difference between the two networks is that recurrent network forms the directed cycles whereas the feedforward networks don't form the directed cycles. Fig 1 represents the different categories of neural network.

The concept of associative memory can be applied to both recurrent and feedforward networks. The associative memory based neural network is mainly used to get the exact recall of the patterns. Initially the original dataset is taken for training the neural network. By applying some translated operations over the training dataset, the output dataset (i.e. test dataset) is computed.

The first Bidirectional Associative Memory (BAM) [1] introduced by Bartkosko in 1988 was the foundation for the later discovered models. BAM is a two layer network works both in forward as well as backward direction. Generally, matrix form of input is taken and exchanged from one neuron to the other in the forward direction whereas the transpose of the input matrix is exchanged in the backward direction. HAMNN is an example of BAM network. A new model of Alpha-beta Bidirectional Associative Memory [2] was developed in 2007 to recognize the finger prints. Alpha is useful in learning the patterns and Beta is useful in recalling the patterns. The basic mathematical operations such as AND (minimum) and OR (maximum) are used to provide the exact recall of stored patterns.

In 2012[3] Mario Aldape-Péreza et al., proposed a concept of Associative Memory Based Classifier (AMBC). AMBC is used for mapping the input and the output pattern. AMBC technique uses two learning phase for strong learning and recalling the stored patterns. The result of the AMBC algorithm is compared with twenty traditional algorithms in Weka. From the experimental results it was found that AMBC algorithm provides best classification results averaged for all datasets. CHAT and CHAT-OHM [4] is proposed using hybrid associative memories to get exact recall on stored patterns. Furao Shen et al., in 2013 proposed a General Associative Memory (GAM) based technique for exact recalling the different types of stored patterns. GAM is used to recall the different types of data such as binary, bipolar, real-valued. GAM technique can be implemented using Auto associative and Hetero associative memory. This consists of three layers: input layer, associative layer and memory layer. The association layer is used to build association between the classes. Memory layer is used to store the input vectors corresponding to the classes. Recurrent Neural Network (RNN) [6] uses the internal memory for storing the input patterns. Back Propagation Through Time (BPTT) is used for training RNN. Use of BPTT fastens the learning process of Neural Network. In 2015, Miguel F. et al., [7] proposed a Morphological Hetero Associative Memory (MHAM) neural network for spinal and disk hernia disease diagnosis. In classical HAM, the weights of the connections are multiplied whereas in MHAM, the weights get added and the result is given as input to the next layer. The main drawback of this technique is that it has a low pattern recall. To provide a high recall of patterns for content retrieval [8], the SUM-OF-SUM and SUM-OF-MAX are used. SUM-OF-SUM and SUM-OF-MAX is the simple matrix operations used for the content retrieval.

III. PROPOSED METHODOLOGY

The proposed methodology consists of three phases: learning phase, recovery phase and calculation of classifier accuracy. Learning phase stores the input patterns in HAMNN. Recovery phase tries to recall the exact stored input patterns from HAMNN using Lern matrix. Finally accuracy of the classifier is calculated by comparing the input matrix (i.e. Preprocessed dataset) used in learning phase and output matrix generated after recovery phase. The activities involved in the methodology are outlined in the Fig. 2.

A. Preprocessing the dataset

Six Healthcare datasets such as Breast Cancer, Cleve, Heart, Hepatitis, Pima and Sick taken from UCI machine learning repository consists of continuous values. So the dataset is given to Weka 3.7 for preprocessing using filters. The range of the tuple values should be taken carefully along with the requirements of the system.

B. Learning phase

Input patterns are stored in the learning phase. To illustrate the working of HAMNN, the sample tuples shown in Table I is considered. Table II shows the Mean vector (1) calculation for the tuples by counting the number of 1's in each tuple.

$$Mean\ vector = \frac{Count\ of\ 1's\ in\ each\ tuple}{Total\ No.\ of\ tuples} \tag{1}$$

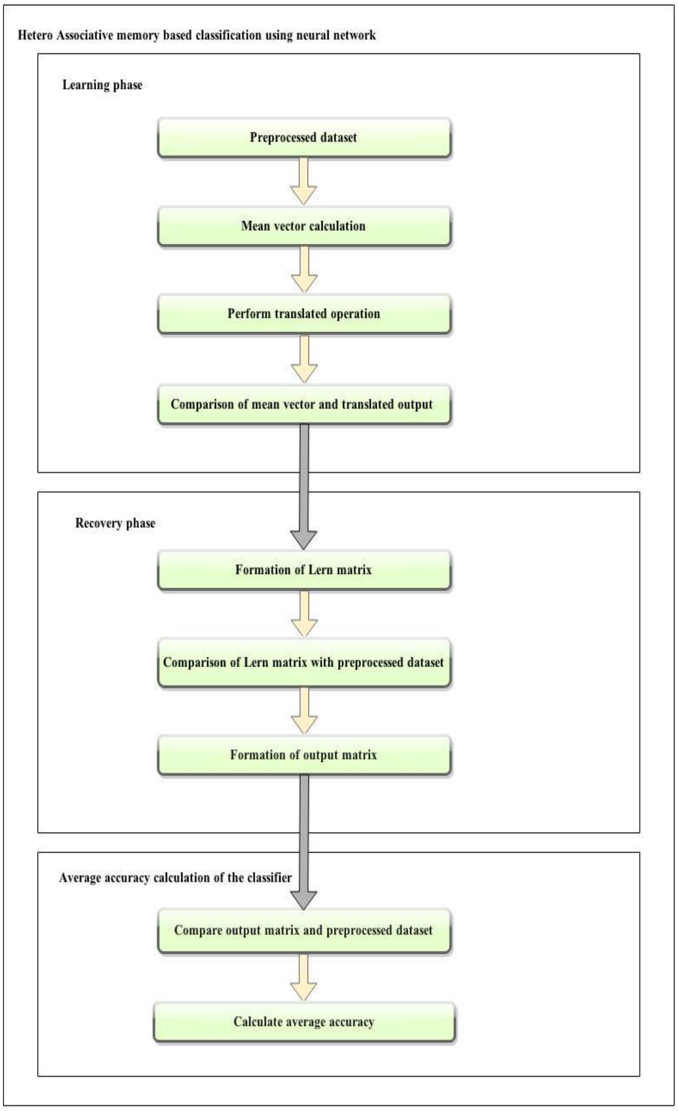

Fig.2 System Flow of the Proposed Methodology

Table I. Sample Tuples

Tuples/Attributes	Att1	Att2	Att3	Att4	Class label
Tuple 1	1	0	0	1	**1**
Tuple 2	0	0	1	0	**0**
Tuple 3	1	1	1	1	**0**
Tuple 4	1	1	1	0	**1**

The translated operation is performed by subtracting sample tuples given in Table I and the mean vector. Translated matrix is given in Table III. Table IV represents the detailed procedure of three phases in algorithmic form.

Table II. Mean Vector

Tuples/ Attributs	Att1	Att2	Att3	Att 4	Clas s label	Mean vector
Tuple 1	1	0	0	1	1	**0.5**
Tuple 2	0	0	1	0	0	**0.25**
Tuple 3	1	1	1	1	0	**1**
Tuple 4	1	1	1	0	1	**0.75**

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table III. Translated Matrix

Tuples/Attributes	Att1	Att2	Att3	Att4	Class label
Tuple 1	0.5	-0.5	-0.5	0.5	**1**
Tuple 2	-0.25	-0.25	0.75	-0.25	**0**
Tuple 3	0	0	0	0	**0**
Tuple 4	0.25	0.25	0.25	-0.75	**1**

Table IV. HAMNN Classifier

Algorithm: HAMNN classifier()
//B[m][n] represents training dataset in binary format, where m represents tuples and n represents attributes
//M[m][1] represents mean vector matrix
//T[m][n] represents translated matrix
//Z[m][n] represents Lern matrix consisting of zeros
//V[m][n] represents resultant output matrix
//A[m][n] represents the resultant matrix after AND operation with class attribute(i.e. last m^{th} attribute)
//Initial parameters to be taken
1) Count_1=0
2) Correct=0
3) Wrong=0
//Learning Phase
4) for (i=1; i<=m;i++)
5) for (j=1;j<=n;j++)
6) Count_1= count of number of 1's in each tuple B[i][j];
7) end for
8) M[i][1]=count_1/m
9) end for
10) for (i=1; i<=m;i++)
11) for (j=1;j<=n;j++)
12) T[i][j] =B[i][j] – M[i][1]
13) end for
14) end for
15) for (i=1; i<=m;i++)
16) for (j=1;j<=n;j++)
17) if (M[i][j]==T[i][j]) then
18) Z[i][j]=1;
19) end if
20) end for
21) end for
//Recovery Phase
22) for (i=1; i<=m;i++)
23) for (j=1;j<=n;j++)
24) if (B[i][j]==Z[i][j]) then
25) V[i][j]=B[i][j] && B[i][n]
26) end if
27) end for
28) end for
//Classifier Accuracy computation
29) for (i=1; i<=m;i++)
30) for (j=1;j<=n;j++)
31) if (B[i][n]==V[i][n]) then
32) Correct=Correct+1;
33) else
34) Wrong=Wrong+1
35) end if
36) end for
37) end for
38) calculate accuracy using Correct and Wrong

C. Recovery phase

The recovery phase recalls the stored patterns used in the learning phase. Lern matrix shown in Table V is computed by the comparing the values of each tuple in Table II with attributes of translated matrix in Table III. From the Table V; it is found that, only the first tuple of the Lern matrix and sample tuples in Table I are similar out of four tuples. If the same process of learning and recovery using Lern matrix is applied for huge number of tuples, the high recall of patterns can be obtained. Further, AND operation is applied between the class label of sample tuples and the attributes of the sample tuples to form the output matrix. Final output matrix is shown in Table VI.

Table V. Lern Matrix

Tuples/Attributes	Att1	Att2	Att3	Att4	Class label
Tuple 1	1	0	0	1	**1**
Tuple 2	0	0	0	1	**0**
Tuple 3	0	0	0	0	**0**
Tuple 4	0	0	0	0	**1**

Table VI. Output Matrix

Tuples/Attributes	Att1	Att2	Att3	Att4	Class label
Tuple 1	1	0	0	1	**1**
Tuple 2	0	0	0	1	**0**
Tuple 3	0	0	0	0	**0**
Tuple 4	1	1	1	0	**1**

D. Accuracy calculation

The input matrix used in the learning phase and output matrix generated after recovery phase are compared for accuracy calculation. Based on the number of similar tuples between the input and output matrix, the accuracy is calculated. The accuracy of the HAMNN classifier is calculated using the measures such as accuracy as in (2) precision as in (3) and recall as in (4).

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

Where, TP is True Positive i.e. positive records classified as positive.
FP is False Positive i.e. negative records classified as positive.
TN is True Negative i.e. negative records classified as negative.
FN is False Negative i.e. positive records classified as negative

IV. EXPERIMENTS AND RESULTS

HAMNN classifier is developed using Lern matrix for six health care datasets such as Breast Cancer, Cleve, Heart, Hepatitis, Pima and Sick from UCI machine learning repository. All healthcare datasets considered in this paper are preprocessed as explained in section III. As a sample, Breast Cancer dataset in binary format after preprocessing is shown in Fig. 3 Using NetBeans IDE 7.4 along with the Encog libraries, HAMNN classifier is built for each dataset for learning and recovering the patterns. Results of the HAMNN classifier over six health care datasets are shown in Table VII. From the results; it is found that HAMNN yields better accuracy and recall for almost all datasets except Sick dataset. It is analyzed that all the attributes of Sick dataset is significant as it cannot be ignored. Hence preprocessing and binarizing the dataset leads to the information loss thus resulting with poor performance. Precision and recall obtained using HAMNN classifier over six healthcare datasets is shown in Fig. 4.

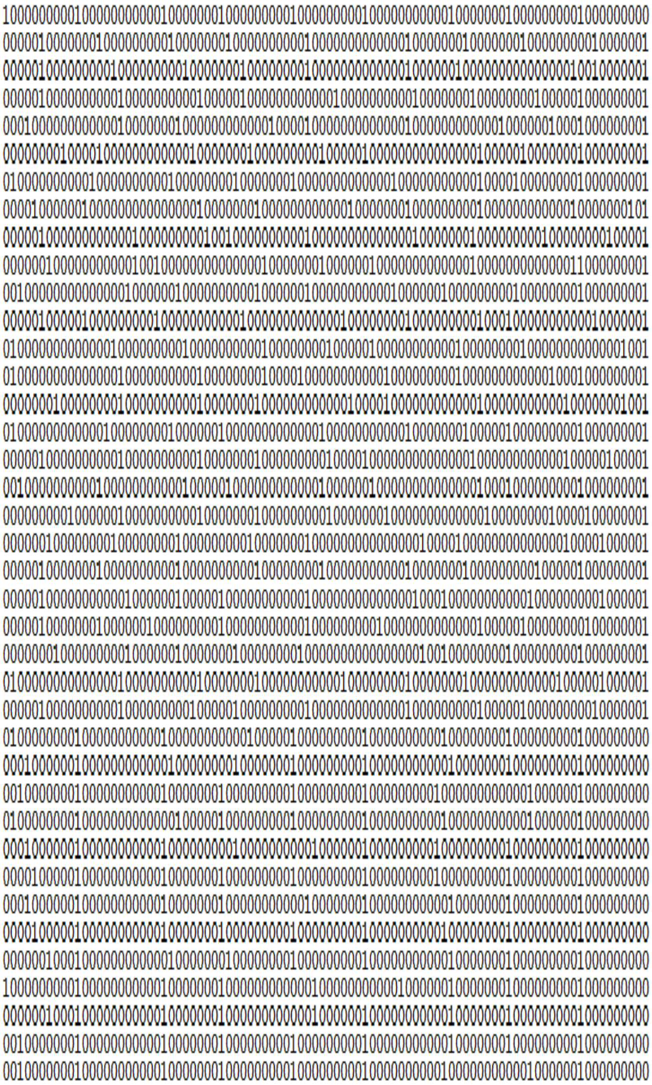

Fig. 3 Breast Cancer Dataset in Binary Format

Table VII. Performance of the HAMNN classifier

DATASET	ACCURACY (%)
Breast Cancer	86.25
Cleve	83.57
Heart	90
Hepatitis	82.5
Pima	90
Sick	45

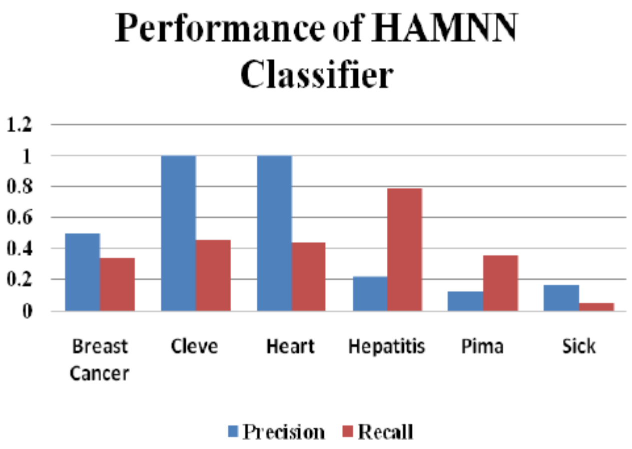

Fig. 4 Comparison of Precision and Recall Obtained for Six Health care Datasets Using HAMNN Classifier

V. CONCLUSION AND FUTURE SCOPE

HAMNN classifier thus built using associative memory and Lern matrix is valuable for disease diagnosis. Use of Lern matrix yields promising accuracy and recall for almost all datasets except Sick. For experimentation, only binary class datasets are considered in this work. In future multi class datasets can be employed for classification along with stronger learning and recovery procedure for the exact recall of the patterns.

REFERENCES

1. Bart kosko, “Bidirectional Associative Memories”, IEEE transactions on systems, man and cybernetics, Vol. 18, No. 1, 1988 49-60.
2. Cornelio Yáñez Márquez and Itzamá López Yáñez, “A New Model of BAM: Alpha-Beta Bidirectional Associative Memories”, Journal of computers, Vol. 2, No. 4, June 2007.
3. Mario Aldape-Péreza, Cornelio Yánez-Márqueza, Oscar Camacho-Nietoa,Amadeo J.Argüelles-Cruz ,“An Associative Memory approach to Medical Decision Support Systems”,Journal of Computer Methods and Programs In Biomedicine,Vol.106, Elsevier Ireland Ltd 2012, 287-307.
4. Uriarte-Arcia AV,Lopez-Ya'nez,Yanez-Marquez C,“One-Hot Vector Hybrid Associative Classifier for Medical Data Classification”,Journal of PLoSONE 9 (4): e95715, doi: 10.1371/journal.pone.009571- 2014.
5. Furao Shen, QiubaoOuyang, WataruKasai , OsamuHasegawa ,“A General Associative Memory Based on Self-Organizing Incremental Neural Network” , Journal of Neurocomputing,Vol.104,Elsevier 2013, 57-71.
6. Gang Bao,ZhigangZeng, “Analysis and Design of Associative Memories Based on Recurrent Neural Network with Discontinuous Activation Functions”, Journal of Neurocomputing ,Vol .77, Elsevier 2012,101-107.
7. Miguel F. Rocha ESIME, Instituto Politécnico Nacional, U.P. A.L. Mateos, G. A. Madero, México D.F. 07738, México, “Implementation of an Expert System to Diagnose Spinal Diseases”,Journal of Industrial and Intelligent Information, Vol. 3, No. 1, March 2015.
8. Zhe Yao, Vincent Gripony and Michael G. Rabbat,“A Massively Parallel Associative Memory Based on Sparse Neural Networks”, Journal of arXiv:1303.7032v2[cs.AI],21 Jul 2013.