A HYBRID NEURO-GENETIC APPROACH FOR EFFICIENT MARBURG VIRUS DISEASE DIAGNOSIS

ISSN: 2141 – 3290
www.wojast.com

AMADIN[1], F. I., EGWUATU[2], J. O. AND EKONG[3*], V. E

[1,2]*Department of Computer Science, University of Benin, Nigeria.*
[3]*Department of Computer Science, University of Uyo, Nigeria.*
[1]*frankamadin@uniben.edu*, [2]*joshegwuatu@gmail.com*,
[3]*victoreekong@uniuyo.edu.ng* **Corresponding Author*

ABSTRACT

Marburg Virus Disease (MVD) is a severe hemorrhagic fever which affects humans and non-human primates. Marburg virus is a genetically unique zoonotic ribonucleic acid virus of the filovirus family. The disease frequently presents multiple signs and symptoms abruptly within two to eight days of infection and has a very short period of attack on a patient. Advances in clinical diagnosis support systems have accelerated the discoveries of artificial intelligent approaches for diagnosis, prognosis and treatments that can support primary care physicians in early detection of potential patients. This paper proposes a hybrid neuro-genetic model driven by a back-propagation neural network and genetic algorithm for the diagnosis of MVD. The system utilizes historical data from Marburg Hemorrhage Fever Documentation (MHFD), analyzes and predicts the risk levels of patients. The Genetic Algorithm (GA) selects optimal features for the Neural Network (NN) training and the developed system bases its prediction solely on the clinical symptoms of MVD and the locations of the patient within the period of incubation of the disease. The proposed neuro-genetic model identified and classified cases with a high accuracy of 97.6% compared to a Back Propagation (BP) NN of 89.8%. This shows that the proposed model can improve the diagnosis comparable to that of the domain experts.

INTRODUCTION

Marburg Virus Disease (MVD) is a severe type of hemorrhagic fever which affects both humans and nonhuman primates. It is caused by a genetically unique zoonotic Ribonucleic acid (RNA) virus of the family of filoviridae. Its recognition led to the creation of this virus family with the five sub-classes of Ebola as the other members of the class. MVD was first recognized in 1967. The patients exhibited the same symptoms that included fever, diarrhea, vomiting, bleeding from many organs, shock and collapse of the circulatory system. In the outbreak 31 persons were infected and seven died (WHO, 2014). The source of the infection was traced to African green monkeys that were imported from Uganda to be used for Polio vaccine research. Many other outbreaks have been reported in Angola (Khristova *et al.*, 2006), Democratic Republic of Congo, Uganda and South Africa (Bausch *et al.*, 2006). The disease can be diagnosed using immuno-histo-chemistry, virus isolation, or Polymerase Chain Reaction (PCR) of blood or tissue specimens from deceased patients (Davis and Stoppler, 2014). These laboratory testing methods may not be applied at the onset of the disease due to the fact that the patient presents many common symptoms with other diseases. The cost and availability of the required equipment makes it more difficult for patients to access these methods. There can be variation observed in disease severity during outbreaks of diseases which may depend on many factors. These factors include quality and availability of medical care to give early diagnosis, infection bouts and route of infection, differences in host population susceptibility (depending on immune and nutritional status and genetics), inherent differences in viral variant virulence and the prevalence of co-infections (particularly malaria and Human immune deficiency virus (HIV) in patients from sub-Saharan Africa) (Bausch *et al.*, 2006; WHO, 2012; Davis and Stoppler, 2014).

In this study we propose a Back Propagation NN based model associated with a GA which aims at improving the prediction accuracy of MVD. The GANN method is applied to MHFD dataset which has been obtained from an MVD data repository from Congo. The dataset consists of 160 subjects.

OVERVIEW OF NEURAL NETWORK AND GENETIC ALGORITHM

Artificial Neural Network (ANN) is an interconnected assembly of simple processing elements, unit or nodes, whose functionality is loosely based on the human brain neuron. The processing ability of the network is stored in the interconnecting unit strengths or weights, obtained by a process of adaptation to or learning from a set of training patterns. ANN is computationally less intensive to suit complex applications though its structure is simple as stated in Liu and Liang (2005). Most ANN applications use Feed Forward (FF) architecture with gradient–based learning like Back Propagation (BP) algorithm (Yao and Liu, 1997) or modified BP algorithm (Hikawa, 2003). As the complexity of a network increases, the search space appears with more and more local optima and gradient–based learning may not always lead to global minima. Moreover BP needs complex operation, which restricts the search coverage. To improve the global convergence capability, an Evolutionary Algorithm (EA) can be used (Rumelhart and McClelland, 1986). Yao (1993) called this special class the "Evolutionary Artificial Neural Networks (EANN's)" in which evolution is another fundamental form of adaptation in addition to learning. EANN can be exploited to design the architecture, learn weight, adapt the learning rule and extract the rule from ANN (Martinetz *et al.*, 1993). EA was broadly classified as Evolutionary Strategies (ES), Evolutionary Programming (EP) and Genetic Algorithms (GA), though many other types have emerged in the recent past (Palmes *et al.*, 2005). The capability of GA in the exploitation of information guides the direction of search towards feasible region and hence it converges at global optima.

Genetic algorithms are search and optimization techniques based on the evolutionary ideas of natural selection and genetics (Goldberg, 1989). They follow the principle of survival of the fittest for better adaptation of species to their environment (Fogel, 1995). Both methods can be combined to get an optimal solution to a problem.

NEURO-GENETIC APPROACH

Neuro-genetic approach is a hybrid of NN and GA. GA and NN may broadly be classified as non–invasive and invasive technique. Non–invasive method combines GA and gradient learning, while invasive method adapts the weight. The noninvasive method evolves the structure of the algorithm. Since it involves gradient method, proper initialization and network implementation is needed to overcome the local minima problem. On the other hand, invasive method uses GA for both weight and topology evolution of ANN. A purely non–invasive approach with a constructive algorithm was demonstrated in Islam *et al.*, (2003) to evolve Cooperative Neural Network Ensembles (CNNE), using incremental learning. This reduced redundancy and maintained diversity to offer a better solution. This improved GA for tuning the structure and parameters of the ANN. Generation of three offspring using different mutation operation led to the improvement. Structural and weight learning by mutation was employed in GNARL algorithm to construct a Recurrent Neural Network (Angeline *et al,* 1994). Based on GNARL algorithm, a Mutation–based Genetic Neural Network (MGNN) is implemented in Palmes *et al.* (2005). In Shanthi *et al.* (2008) a Neuro-Genetic approach was proposed for feature selection in the diagnosis of stroke disease. Here multilayer perceptron was used whose inputs are automatically selected using GA and the experimental results show better classification accuracy with fewer inputs as features. In a study by Elveren and Yumusak (2009), tuberculosis diagnosis was successfully achieved using a multilayer neural network (MLNN) with two hidden layers and GA for network training and feature selection.

MATERIALS AND METHODS

The symptoms of Marburg virus disease were obtained from the documentation of Marburg Hemorrhagic Fever (MHFD) in Durba and Watsa, Democratic Republic of the Congo. It is a Clinical Documentation of Features of Illness and Treatment (CDFIT) that was given to 160 patients of Marburg virus disease during the outbreak in democratic republic of Congo.

Dataset

The dataset is comprised of different symptoms of MVD at various infestation stages. The infestation stages in are: early symptoms (median onset of symptoms: 1 – 2 days), hemorrhagic manifestation (5 – 8 days) and terminal symptoms (more than 9 days). Table 1 shows these symptoms and their categorization. Twenty-eight symptoms are categorized into three groups: Fever, General and Hemorrhage,

Table 1: Symptoms categorization of MVD

S/N	Category	Symptoms	Labels	Manifestation
1	Fever	High Fever	S_1	A
2		Severe Headache	S_2	V
3		Fatigue	S_3	A
4	General	Loss of appetite	S_4	V
5		Vomiting	S_5	V
6		Generalized pain	S_6	V
7		Diarrhea	S_7	V
8		Dyspnea	S_8	C
9		Abdominal pain	S_9	C
10		Sore throat	S_{10}	C
11		Hiccups	S_{11}	C
12		Conjunctivitis	S_{12}	C
13		Chest pain	S_{13}	U
14		Lumbar pain	S_{14}	U
15		coughing	S_{15}	U
16		Coma >24 h	S_{16}	R
17		Dysphagia	S_{17}	C
18		Nausea	S_{18}	A
19		Melena	S_{19}	C
20		Epistaxis	S_{20}	U
21		Bleeding at injection sites	S_{21}	U
22	Hemorrhage	Hemoptysis	S_{22}	R
23		Petechiae	S_{23}	R
24		Vaginal bleeding	S_{24}	R
25		Hematuria	S_{25}	R
26		Hematemesis	S_{26}	C
27		Bloody diarrhea	S_{27}	C
28		Bleeding gums	S_{28}	C

[R=Rear, C= Common, U= Uncommon, V= Very common, A= Almost all]

In Table 2, the weighting scores allotted to different manifestation degrees ranging from 1 to 5 are indicated. A symptom with a weight of 1 may not manifest at all or may have low manifestation degree.

Table 2: Manifestation weighing of various symptoms

Manifestation	Meaning	Label	Assigned
< 10%	Rear	R	1
10% to 34%	Uncommon	U	2
35% to 59%	Common	C	3
60% to 79%	Very common	V	4
80% to 99%	Almost all	A	5

Optimization of Symptoms Using GA

All the twenty eight symptoms may not manifest at the onset of the disease. The problem of determination of related symptoms for MVD can be formulated into a set of mathematical equations. The set of symptoms (S) is given by Equation (1).

$$S=\{S_1, S_2, \ldots S_n\} \quad (1)$$

If *d* is the degree of each symptom of the disease and *A* is the disease, the following indices can be formulated:

i = 1 to n (where I is the disease index)
j = 1 to n (where j is symptom index)
k = 1 to n (manifestation degree)
t = 1 to 14 (period of manifestation)
n = number of symptoms.

With GA our objective is to minimize the number of symptoms used for the prediction of the MVD. The objective function for this will be given as a quadratic error (cost) function which is derived from the mean weighted scores of the various symptoms given in Equation (2):

$$f(w_e) = 1/n \sum_{j=1}^{n} (w_t - w_j)^2 \quad (2)$$

Where w_t is a constant, the expected weighting score for a determinant symptom w_j and n the number of symptoms.

Equation (2) can be expanded to give Equation (3) while the error cost function can be written as Equation (4).

$$(w_t - w_j)^2 = {w_t}^2 - 2w_t w_j + {w_j}^2 \quad (3)$$

$$f(w_e) = \frac{1}{n} \sum_{j=1}^{n} |({w_t}^2 - 2w_t w_j + {w_j}^2)| \quad (4)$$

If n = 1, Equation (4) becomes Equation (5)

$$f(w_e) = {w_t}^2 - 2w_t w_j + {w_j}^2 \quad (5)$$

Equation (5) is subjected to the following constraints (C1 and C2:

C1: each of the symptoms must have high weighted score $3 < w_{jk} \leq 5$

C2: the highest manifestation degree is 100% and corresponds to $w_{jk} = 5$

Other components of the GA are specified in Table 3. Each chromosome is evaluated using the function shown in Equation (6).

$$F = 1/(1 + f(w_e)) \quad (6)$$

To obtain the fitness value of each chromosome in the set of symptoms we calculate the average fitness of an individual chromosome using Equation (7) and Probability of the chromosome to be selected using Equation (8).

$$f_{Avg} = (\sum_{j=1}^{n} f_j)/n \quad (7)$$

$$p_j = f_j/f_{Avg} \quad (8)$$

The summary of the chromosome selection is shown in Table 4.

Table 3: Genetic algorithm component specification

Component	**Value**
Search method	Genetic Algorithm
Population size	28
Encoding	Binary
Evaluation	Fitness function
Selection	Rhowlettete wheel
Cross over function	Single point
Flip rate	0.01
Stoppage criteria	Till convergence to best solution is observed
Generation	50

Table 4: Summary of Chromosome Selection of Initial Population

Chromosome ID	Weighting score (w_j)	Chromosome	Fitness (F) $F = 1/(1 + f(w_e))$	Probability (p) $p = f_i/f_{Avg}$	Expected count
1	5	00000	1.0000	3.0127	3
2	4	00001	0.5000	1.5063	2
3	5	00000	1.0000	3.0127	3
4	4	00001	0.5000	1.5063	2
5	4	00001	0.5000	1.5063	2
6	4	00001	0.5000	1.5063	2
7	4	00001	0.5000	1.5063	2
8	3	00100	0.2000	0.6025	1
9	3	00100	0.2000	0.6025	1
10	3	00100	0.2000	0.6025	1
11	3	00100	0.2000	0.6025	1
12	3	00100	0.2000	0.6025	1
13	2	01001	0.2000	0.6025	1
14	2	01001	0.1000	0.3013	0
15	2	01001	0.1000	0.3013	0
16	1	10000	0.0588	0.1771	0
17	3	00100	0.2000	0.6025	1
18	5	00000	1.0000	3.0126	3
19	3	00100	0.2000	0.6025	1
20	2	01001	0.1000	0.3013	0
21	2	01001	0.1000	3.0127	3
22	1	10000	0.0588	0.1771	0
23	1	10000	0.0588	0.1771	0
24	1	10000	0.0588	0.1771	0
25	1	10000	0.0588	0.1771	0
26	3	00100	0.2000	0.6025	1
27	3	00100	0.2000	0.6025	1
28	3	00100	0.2000	0.6025	1

From Table 4, the initial summary of the initial population are given as f_t =8.394, f_{avg} = 0.3319 and N=28.0. The expected count shows the number of a chromosome that can be selected for reproduction of offspring that will make up the population for the next generation. The integer part of the expected fitness determines the number of the chromosome that will be selected while the fractional part shows its chances of being repeated. Figure 1 shows the graphical representation of the expected counts of the various symptoms used in the GA.

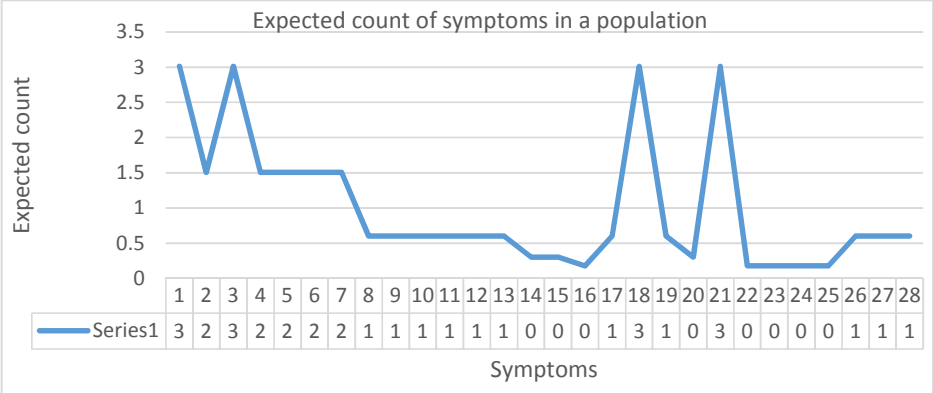

Figure 1: Graph of expected count for a symptom in the population

The GA produces the best combination of input features to provide the solution with less computational complexity but more accuracy. The selected attributes using GA are shown in Table 5.

Table 5: Optimal Input Parameters

S/N	Symptom	Symptom code
1	Fever	S_1
2	Headache	S_2
3	Fatigue	S_3
4	Loss of appetite	S_4
5	Vomiting	S_5
6	Diarrhea	S_7
7	General Pain	S_6
8	Bleeding at injection sites	S_{21}

GANN Architecture

The summary of processes for evolving optimal symptoms for the diagnosis of MVD disease is shown in Figure 2.

EXPERIMENT AND RESULTS

The system was implemented with MATLAB 7.7.0 (R2008b). The NN and global optimization toolboxes were deployed in the system in simulating the GANN model. The system used seventy percent of the data (112 samples) for training. Testing and evaluation were carried out with 24 records (15%) each. In every training session, GA selects training samples randomly from the entire dataset thereby generating different values of mean square error (MSE).

Neural Network training

The symptoms in Table 5 are the major input parameters of the NN. The input parameters are represented by P. P is a set of symptoms that correspond to an output as derived from Equation (8):

$$P_j=\{p_1, p_2, \ldots p_n\}$$

Where P_j represent the j^{th} parameter and n is the total number of parameters. The set of corresponding MVD states which is modeled as the target scales T is given as follows:

$T = \{MVD, RG4AD, NRG4AD, ND\}$

where, MVD = Marburg virus disease, RG4AD = Risk Group 4 Agent related disease, NRG4AD = Non Risk Group 4 Agent related disease and ND = No disease.

The set in T is encoded into binary values in equation as follows:

$T = \{1000, 0100, 0001, 0000\}$

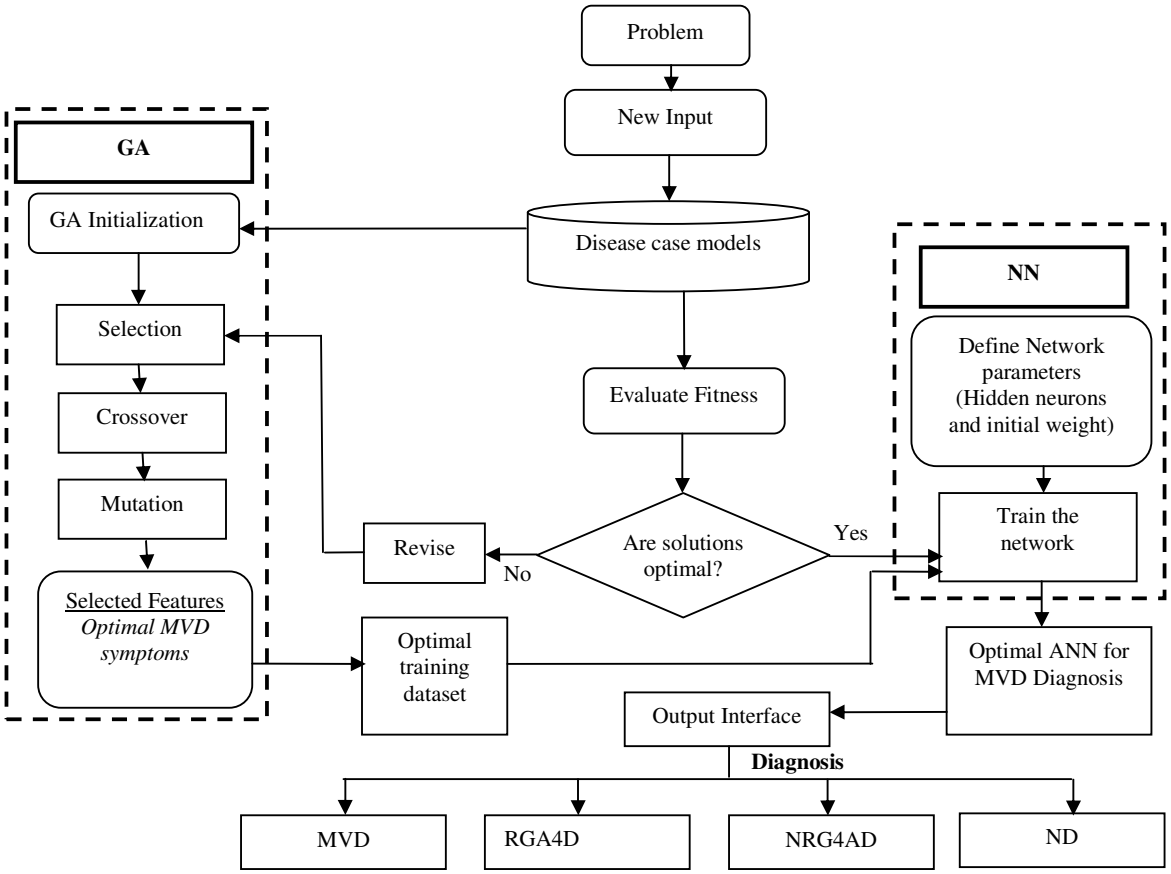

Figure 2: GANN process for the diagnosis of MVD

The input data are preprocessed and are encoded within the range -1 to 1 and most of the inputs are of two-states. The data is partitioned and Table 6 shows the number of records in the training set, validation set and test set.

Table 6: Data partition for training, validation and test

S/N	Data partition set	Record	Percentage (%)
1.	Training	112	70.0
2.	Validation set	24	15.0
3.	Test set	24	15.0
4.	Ignored set	0	0.0
	Total	160	100.0

The NN is a 3-layered feed forward architecture with sigmoid function for neuron activation. The input layer has eight neurons corresponding to the inputs, the hidden layer consists of seven neurons and the output layer has 4 neurons. The first eight input nodes represent the symptoms for the diagnosis of MVD. The output layer consists of four output nodes that represent the predicted disease. Optimal weights of the NN were generated via the GA in four stages: Initialization, selection, crossover and mutation (Figure 2). The network is trained using Back-propagation algorithm. The error rates on the training and test data are shown in Table 7.

Table 7: Error rates on training of the network

Method	**Training**	**Testing**
Correlation coefficient	0.9967	0.9989
Mean absolute error	0.0049	0.0017
Root mean square error	0.0402	0.0195
Relative absolute error	1.0659	0.399
Root relative absolute error	8.3576	2.8159

To evaluate the performance of the GANN model with a given set of operators and parameter settings, we performed a series of independent runs with the test data using a BP NN, recording the evaluation of the best individual as a function of the number of iterations. Afterwards, we averaged the results so as to reduce the variations introduced by the stochastic nature of the BP NN algorithm. Table 8 shows the average prediction accuracy of Neuro-Genetic approach and the BP NN approach. The average predictive accuracy is the corresponding percentage values of the regression values of the training, validation and test values. The result shows clearly that the hybrid GANN system predicts better than the BP NN with an accuracy of 97.6% and 89.8% respectively. This accuracy is as a result of the reduction in the input parameters obtained by the GA and thus eliminated the chances of removing vital parameters.

Table 8: Average Predictive Accuracy

Approach	Training	Validation	Testing
Neural Network	78.2%	80%	89.8%
Neuro-Genetic Approach	86.9%	81.1%	97.6%

CONCLUSION

Currently, Africa is recovering from the worst outbreak of hemorrhagic fever caused by Ebola virus. Already infected people have been diagnosed and quarantined. The goal now is to limit the disease spread. This paper demonstrates a practical application of an intelligent CDSS in the health sector by presenting a hybrid Neuro-Genetic approach in the selection of optimal vital input features that will predict the target disease in a patient suspected to have MVD. Experimental results showed that the system can give better prediction accuracy that will support medical expert decisions in the diagnosis of a lethal hemorrhagic fever caused by Marburg virus. This will reduce the risk level of the disease in the event of an outbreak.

REFERENCES

Angeline, P. J., Saunders, G. M. and Pollack, J. B. (1994). An evolutionary algorithm that constructs recurrent neural networks. *IEEE Trans. on Neural Networks*, 1(5):54–64.

Bausch, D.G., Nichol S.T., Muyembe-Tamfum J.J., Borchert M., Rollin P.E., Sleurs H., Campbell P., Tshioko F.K., Roth C. and Colebunders R., (2006) Marburg hemorrhagic fever associated with multiple genetic lineages of virus. *N. Engl. J. Med*. 2006, 355:909–919.

Davis, C. P and Stoppler, M. C (2014) Marburg virus history, symptoms and treatment, United States Centers for Disease Control and Prevention. Marburg Hemorrhagic Fever (Marburg HF), Available on line at: http://www.cdc.gov/vhf/marburg/index.html (Accessed 23 January 2016).

Elveren E. and Yumuşak N. (2009) Tuberculosis Disease Diagnosis Using Artificial Neural Network Trained with Genetic Algorithm, *Journal of Medical Systems*, 35(3):329-332.

Fogel, B. D. (1993). Using Evolutionary Programming to Create Neural Networks that are Capable of Playing Tic-Tac-Toe, in: *International Conference of Neural Networks*, 2:875-880, San Francisco, IEEE.

Goldberg, E. D., (1989). Zen and the Art of Genetic Algorithms, in: *Proceedings of the Third International Conference on Genetic Algorithms*, pp. 80-85, Morgan Kaufmann.

Hikawa, H. (2003). A Digital hardware pulse–mode neuron with piecewise–linear activation function. *IEEE Transactions on Neural Networks*. 14(5):1028–1037.

Islam, M., Yao X., and Murase, K. A. (2003). Constructive Algorithm for training cooperative neural network ensembles. *IEEE Trans. on Neural Networks*, 14(4):820–834.

Khristova, M. L., Towner J. S., Sealy T. K., Vincent M. J., Erickson B. R., Bawiec D. A., Hartman A. L., Comer J. A., Zaki S. R., and Ströher U. (2006). Marburgvirus genomics and association with a large hemorrhagic fever outbreak in Angola. *J. Virol.* 80:6497–6516.

Liu, J., and Liang D. (2005). A Survey of FPGA–Based Hardware Implementation of ANNs. *Proc. IEEE Conf.,* pp. 915–918.

Martinetz, T. M., Berkovich, S. G., and Schulten, K. J. (1993). Neural–gas Network for Vector Quantization and its Application to Time–series Prediction. *IEEE Trans. on Neural Networks*, 4:558–569.

Palmes, P. P., Hayasaka T., and Usui, S. (2005). Mutation–Based Genetic Neural Network. *IEEE Trans. on Neural Networks, 2005*, 16(3):587–600.

Rumelhart, D., and McClelland, J. (1986). *Parallel distributed processing: Explorations in microstructure of cognition*. Cambridge, MA: MIT Press.

Shanthi D., Sahoo G., and Saravanan N. (2008). Input Feature Selection using Hybrid Neuro-Genetic Approach in the Diagnosis of Stroke Disease, *IJCSNS International Journal of Computer Science and Network Security*, 8(12):99-107.

WHO, (2012). Marburg Hemorrhage fever, Available online at: http://who.int/csr/disease/marburg/en (Accessed 23 January 2016).

WHO, (2014). Marburg virus disease, Available online at: http://who.int/csr/disease/marburg/en (Accessed 23 January 2016).

Yao, X., and Liu Y. (1997). A New evolutionary system for evolving artificial neural networks. *IEEE Transactions on Neural Networks*, 8(3):694–713.

Yao, X. (1993). A review of Evolutionary Artificial Neural Networks. *Int. J. Intell. Syst*., 8(4): 539–567.