International Journal of Computer Applications (0975 - 8887)
Volume 134 - No.6, January 2016

A Comparison of Grammatical Bee Colony and Neural Networks in Medical Data Mining

Tapas Si
Assistant Professor
Department of Computer Science & Engineering
Bankura Unnayani Institute of Engineering
Bankura, West Bengal, India

Sk. Sujauddin
Lead - Quality Assurance
Mindteck India Ltd.
Millennium City Information Technology Park, Tower II
Sector V, Salt Lake, Kolkata 700091, India

ABSTRACT

This paper proposes a novel application of Grammatical Bee Colony for classification of medical data. Grammatical Bee Colony is a Swarm Programming algorithm generally used for automatic computer program generation in any arbitrary language. In this paper, Grammatical Bee Colony based classifier is designed and applied in medical data mining. The proposed method is applied on ten medical data sets and obtained results are compared with Multi-Layer Perceptron classifier trained with Levenberg-Marquardt algorithm. The proposed method statistically outperforms other method.

General Terms

Medical Data Mining, Machine Learning

Keywords

Medical data mining, Machine learning, Classification, Swarm programming, Grammatical bee colony, Artificial neural network

1. INTRODUCTION

Medical data mining is an important research area in medicine during past several years. Classification is one of the major clinical tasks in diagnosis of new disease. And classification tasks are also useful in the study of the different patterns in the medical data. The objective of classification of medical data is to predict the diagnostic decision (*positive or negative*). Artificial Neural Network (ANN) is a much popular machine learning tool and it is widely used in diagnostic classification of patients [1, 2]. ANN can handle diverse types of medical data and integrate them into categorized outputs [2]. In this paper, Grammatical Bee Colony (GBC) algorithm is used in classification of well-known medical data. T. Si et al. [3] proposed GBC algorithm and it is a *Swarm Programming*(SP) algorithm, generally used to generate computer programs in any arbitrary language. In GBC algorithm, Artificial Bee Colony (ABC) is used as a learning algorithm in mapping from *genotype* (integer codons) to *phenotype* (computer program) while evolving the computer programs. In this paper, a novel application of GBC is presented as a binary classifier for medical data. The proposed GBC classifier is applied on ten medical data sets and a comparative study has been made with Multi-Layer Perceptron (MLP) trained using Levenberg-Marquardt (LM) method [2]. The GBC classifier statistically outperforms MLP classifier.

2. PROPOSED GBC CLASSIFIER

GBC is a Swarm Programming algorithm, and it is used in automatic program generation in any arbitrary language. ABC algorithm is used as a search engine in GBC. The computer programs are evolved using Backus-Naur Form (BNF) of Context-Free Grammar (CFG) through genotype-to-phenotype mapping. Genotype is the set of integer codons i.e food source's position and phenotype is the evolved computer program. The readers are encouraged to go through the paper [3] for details of GBC algorithm. The preliminary steps of the GBC classifier is the defining appropriate Context-Free Grammar in BNF for the data set to be classified. The role of GBC is to evolve computer program that computes a function $Y = f(X)$ where X is the set of input attributes in the data set. The class label is determined using a logistic sigmoid function followed by a threshold function. The GBC classifier is depicted in Fig. 1. In next, Context-Free Grammar in BNF is described.

2.1 Context-Free Grammar

Context-Free Grammar in BNF is used in *genotype-to-phenotype mapping*. BNF of CFG for generating computer program in MATLAB language is described below:

```
1. <EXPR> :=  <OP> (0) | <VAR> (1)
2. <OP> :=    plus(<EXPR>,<EXPR>) (0)
            | minus(<EXPR>,<EXPR>) (1)
            | times(<EXPR>,<EXPR>) (2)
            | pdivide(<EXPR>,<EXPR>) (3)
            | abs(<EXPR>) (4) | sqrt(<EXPR>)(5)
            | log2(<EXPR>)(6) | exp(<EXPR>)(7)
            | sin(<EXPR>)(8)  | cos(<EXPR>)(9)
3. <VAR> := x1 (0) | x2 (1)|...| xn (n-1)
```

pdivide() is the protected division to avoid 'division-by-zero' error and x1, x2, ..., xn are the input attributes in the data sets.

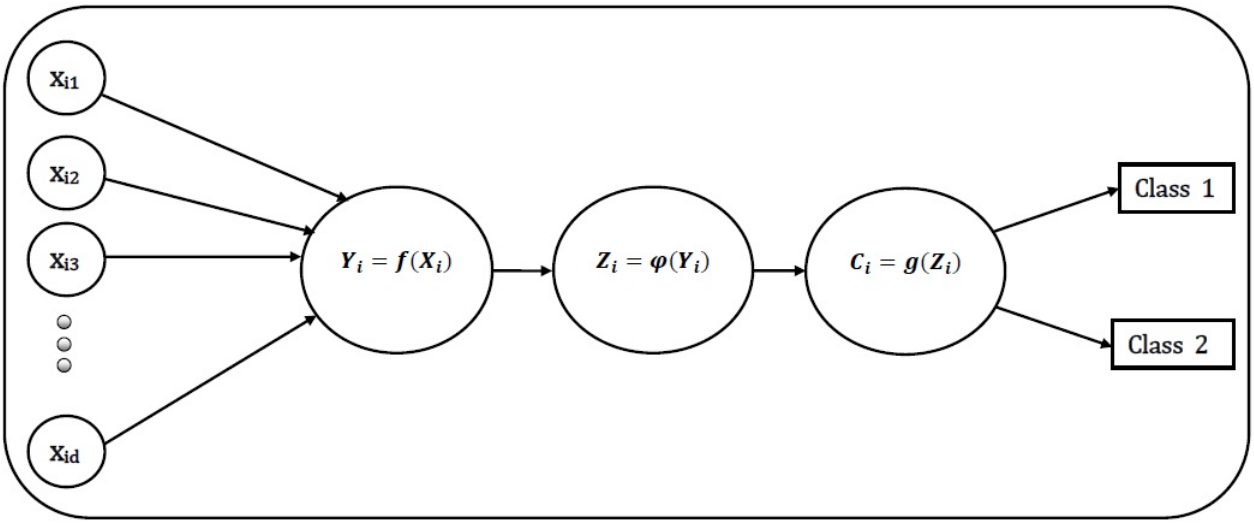

Fig. 1. GBC classifier.

2.2 Genotype-to-Phenotype Mapping

The food source position are generated in the range $[0, 255]$ and rounded-off to its nearest integer values to form the codons. For an example, a part of genotype is given in Fig. 2.

160	140	172	192	55	50	205	62	11	75

Fig. 2. An example of a part of genotype

A *mapping process* is used in mapping from integer-valued food source position to the rule number in the derivation using CFG by the following ways:
Rule=(codon) MOD (number of choices for the current non-terminal)
If the current non-terminal <EXPR> is in derived string, then the rule number is generated as follows:
Rule=(160 mod 2)=0
<EXPR> will be replaced by <OP>.
A complete derivation of phenotype from genotype in Fig. 2 is given below by assuming that the data set has $n = 10$ attributes:

```
<EXPR>
:=<OP>                                  (160 mod 2)=0
:=plus(<EXPR>,<EXPR>)                   (140 mod 10)=0
:=plus(<OP>,<EXPR>)                     (172 mod 2)=0
:=plus(times(<EXPR>,<EXPR>),<EXPR>)(192 mod 10)=2
:=plus(times(<VAR>,<EXPR>),<EXPR>) (55 mod 2)=1
:=plus(times(x1,<EXPR>),<EXPR>)     (50 mod 10)=0
:=plus(times(x1,<VAR>),<EXPR>)      (205 mod 2)=1
:=plus(times(x1,x3),<EXPR>)         (62 mod 10)=2
:=plus(times(x1,x3),<VAR>)          (11 mod 2)=1
:=plus(times(x1,x3),x6)             (75 mod 10)=5
```

When derivation process runs out of codons, *wrapping* is carried out once.

2.3 Classification

The evolved computer programs are the functions $Y_i = f(X_i)$ discovered from the data set X where $i = 1, 2, 3, ..., M$. After evaluating the function $f(X_i)$, the output Y_i for i^{th} input X_i is used to predict the class label $C_i, C_i = \{1, 2\}$. A sigmoid function $\varphi(Y_i)$ is used in mapping from Y_i to $Z_i \in (0, 1)$ as follows:

$$Z_i = \varphi(Y_i) = \frac{1}{1 + e^{-Y_i}} \tag{1}$$

Finally, class label C_i is assigned to the input X_i using a threshold function $g(\cdot)$ as following:

$$C_i = g(Z_i) = \begin{cases} 1 & Z_i < 0.5 \\ 2 & Z_i \geq 0.5 \end{cases} \tag{2}$$

$C_i = 1$ is positive class and $C_i = 2$ is negative class in medical data set.

2.4 Fitness Function

In GBC algorithm, two objective functions are used such as misclassification rate (MR) and geometric mean(GM) [6] over training data. GM measures the trade-off between the sensitivity and specificity by the following:

$$GM = \sqrt{sensitivity \times specificity} \tag{3}$$

A classifier is said to be good classifier if it provides higher classification accuracy and a good trade-off between sensitivity and specificity. The first objective function misclassification rate is to be minimized whereas second objective function geometric mean is to be maximized. The misclassification rate, sensitivity, specificity and GM values are measured using confusion matrix [5]. The two conflicting objective functions are transformed into a single objective function to be minimized as following:

$$F = w_1 \times MR + w_2 \times (1 - GM) \tag{4}$$

Where w_1,w_2 are the weighting factor and $w_1 = w_2 = 0.5$. The objective function F is used as fitness function in GBC algorithm.

3. EXPERIMENTAL SETUP

3.1 Parameter Settings

In GBC, the parameters are set as following: population size=100, dimension=100, limit=10, maximum number of cycles=500. In LM algorithm, the maximum number of epochs is 2000 and threshold error is set to 0.001, initial $\mu = 0.001$, μ decrease and increase factors are 0.1 and 10 respectively, maximum $\mu = 1e10$.

3.2 PC Configuration

(1) Operating System: Windows 7
(2) CPU: AMD FX -8150 Eight-Core 3.6 GHz
(3) RAM: 16 GB
(4) Software: Matlab 2010b

4. RESULTS & DISCUSSION

The proposed Grammatical Bee Colony based classifier is applied on ten medical data sets such as Cleveland heart, liver disorders (BUPA), PIMA Indians Diabetes, Wisconsin diagnostic breast cancer (WDBC), Wisconsin prognostic breast cancer (WPBC), Lung Cancer, Vertebral column, Echocardiogram (ECG), SPECT heart, SPECTF heart. These data sets are collected from UCI Machine Learning Repository [4]. A short description of the data sets is given in Table 1 and the detail description is available from the Ref. [4]. The missing values in the data are replaced by *attribute mean value* [5] and the data are normalized in the range $[0, 1]$. K–fold Cross–validation [5] is used for estimating generalization error of the proposed classifier and the value of K is 10 in this work. The performance of GBC classifier is compared with MLP trained with LM algorithm (MLP-LM).

Table 1. Characteristic of Data sets

Data Set	Number of patterns	Number of features	Number of classes
Cleveland Heart	303	14	2
Liver Disorder	345	7	2
PIMA Diabetes	768	8	2
WPBC	198	34	2
WDBC	569	32	2
Vertebral	310	6	2
ECG	131	13	2
SPECT	267	22	2
SPECTF	267	44	2
Lung Cancer	32	57	3

The mean and standard deviation of training accuracies and mean of CPU time for training are given in Table 2. The mean and standard deviation of testing accuracies are given in Table 3.
Wilcoxon Signed Ranks Test [7] is carried out using mean testing accuracy values to compare the performance of the GBC classifier with MLP-LM classifier. The sum of positive ranks $\sum R^+$ is 53.00 and sum of negative ranks $\sum R^-$ is 2.00. The obtain $p - value$ is 0.009344 which is lower than the significance level 0.01. This $p-value$ is also much lower than the significance level 0.05. From this analysis, it has been seen that the proposed classifier statistically outperforms MLP-LM classifier. Though it is observed from Table 1 that training performance of MLP-LM classifier is higher than that of GBC classifier. To compare the class wise classification accuracy of data, sensitivity, specificity and GM measure are

Table 2. Mean and standard deviation of training accuracy and mean CPU time for training.

Data Set	CGBC		MLP-LM	
	Mean±std.	Time	Mean±std.	Time
Cleveland Heart	85.00± 0.70	13.33	99.60±0.32	0.30
Liver Disorder	68.71± 1.08	13.03	92.09± 0.43	0.93
PIMA Diabetes	75.24± 0.50	12.48	92.74±0.38	1.86
WPBC	75.78± 0.52	12.11	100.00±0.00	0.14
WDBC	93.84± 0.37	13.89	100.00±0.00	0.32
Vertebral	80.47± 1.69	10.48	42.64± 0.64	0.96
ECG	87.08±0.42	11.69	92.42± 0.83	2.50
SPECT	82.67± 0.69	13.16	94.16± 0.17	0.06
SPECTF	79.05± 1.02	14.29	100.00± 0.00	0.57
Lung Cancer	97.88± 0.79	12.55	100.00± 0.00	0.43

Table 3. Mean and standard deviation of testing accuracy.

Data Set	CGBC	MLP-LM
Cleveland Heart	79.81 ±0.63	73.63 ± 4.26
Liver Disorder	68.25± 2.94	67.17± 4.97
PIMA Diabetes	77.47± 4.05	69.11 ±1.80
WPBC	70.91± 1.10	65.95 ±6.29
WDBC	92.06±1.38	94.48± 1.53
Vertebral	76.97 ±3.38	39.81± 1.96
ECG	84.84±4.19	77.81 ±4.55
SPECT	79.05± 2.26	71.90±2.49
SPECTF	67.68±2.24	63.01± 3.31
Lung Cancer	90.37± 5.81	55.79± 12.59

used and tabulated in Table 4. GM measures the trade-off between the sensitivity and specificity. The GBC classifier performs better than MLP-LM classifier in class wise classification for all except WDBC data set. The computation time of GBC classifier is higher in training than that of MLP-LM classifier. The derivation of phenotype from the genotype of each individual in GBC with one time wrapping takes higher computational time. It is also observed that all the codons in genotype are not used during the derivation process. Derivation stops when all the symbols in the derived string are terminals. Some codons are left as unused. Optimal use of genotype's length can reduce the computational time of GBC for training.
The GBC evolved programs for each fold with training and testing accuracy in classification of Lung cancer data are given in Table 5. From this table, it is observed that GBC evolves different programs for different folds. It is also observed that all the 56 attributes in Lung cancer data set are not used in the evolved program for the same fold. These different programs represent different functional relationships of output class with the input attributes in data set. From this discussion, it may be concluded that GBC is able to discover the knowledge about the relationship of patterns with its associated classes in the data set.
The above analysis of results demonstrates that the proposed GBC classifier is effective in classification of medical data. It also efficient in knowledge discovery from the medical data.

5. CONCLUSIONS

A novel application of Grammatical Bee Colony in medical data mining is proposed in this paper. The Grammatical Bee Colony classifier applied for classification of medical data sets and obtained results are compared with MLP trained using LM method. The pro-

Table 4. Mean sensitivity, specificity and GM.

Data Set	CGBC			MLP-LM		
	Sensitivity	Specificity	GM	Sensitivity	Specificity	GM
Cleveland Heart	74.43	84.39	79.17	67.65	78.73	72.95
Liver Disorder	73.50	61.03	66.94	72.46	59.82	65.79
Diabetes	71.46	80.69	75.91	55.46	76.47	65.10
WBCP	77.58	49.53	61.87	72.83	44.16	56.53
WBCD	91.56	92.89	92.21	97.66	89.14	93.28
Vertebral	82.33	74.42	78.23	95.71	0.00	0.00
ECG	71.02	91.50	80.49	68.52	82.52	75.16
SPECT	83.57	61.85	71.24	77.52	50.10	62.27
SPECTF	71.11	54.65	62.05	68.07	43.29	54.25
Lung Cancer	88.92	98.00	93.24	62.91	20.33	31.03

Table 5. GBC evolved programs of each fold with training and testing accuracy for Lung cancer.

Fold#	Program	Training accuracy	Testing Accuracy
1	`times(x48,pdivide(minus(x24,pdivide(cos(exp(sin(abs(sqrt(x27))))) ,plus(pdivide(exp(x4),x14),minus(x26,x4)))),plus(sqrt(x13),x27)))`	96.43	100.00
2	`plus(exp(cos(pdivide(exp(sin(x33)),log2(x20)))), pdivide(minus(x23,x49),x21))`	96.49	100.00
3	`times(x32,minus(x26,times(x54,x1)))`	97.65	90.91
4	`log2(plus(plus(x56,abs(minus(x21,x10))),times(x8,x45)))`	98.23	93.33
5	`sin(cos(log2(times(plus(abs(minus(x34,x23)),x1),pdivide (times(x40,times(plus(x20,exp(abs(x43))),x31)),x39)))))`	98.59	83.33
6	`abs(pdivide(x56,x13))`	98.26	85.00
7	`plus(sin(abs(exp(log2(minus(cos(x38),x3))))),cos(minus(x31,minus (pdivide(x56,x23),sin(csqrt(x31))))))`	98.52	86.36
8	`minus(plus(x51,exp(cos(x44))),plus(abs(cos(plus(x22,pdivide (plus(abs(x50),x51),sin(pdivide(abs(x13),sin(x56)))))))),x3))`	98.27	88.00
9	`minus(x32,minus(sin(x31),minus(x13,minus(abs(minus(x30, exp(plus(minus(x14,pdivide(x17,x18)),sin(log2(plus (x2,pdivide(minus(x19,x49),x50)))))))),x56)))`	98.08	89.29
10	`cos(sqrt(minus(x32,minus(sqrt(minus(x13,x33)),x36))))`	98.26	87.50

posed GBC classifier statistically outperforms MLP. The computational time of GBC classifier for training is higher than that of MLP. The performance improvement of the proposed GBC classifier will be the future work of this paper. In this paper, GBC algorithm is used for binary classification of medical data. This work can also be extended by the development of multi-class classifier using GBC algorithm.

6. REFERENCES

[1] J. Iavindrasana, G. Cohen, A. Depeursinge, H. Muller, R. Meyer and A. Geissbuhler, Clinical Data Mining: A Review, *IMIA Yearbook of Medical Informatics*, 2009.

[2] F. Amato et al.: Artificial neural networks in medical diagnosis, *J Appl Biomed.*, Vol. 11, pp. 47–58, 2013. DOI:10.2478/v10136-012-0031-x

[3] T. Si, A. De, A.K. Bhattacharjee: Grammatical Bee Colony, *In: B.K. Panigrahi et al. (Eds.): SEMCCO 2013, Part I, LNCS 8297*, 2013, pp. 436–445

[4] http://cml.ics.uci.edu.

[5] L. Han, M. Kamber: Data Mining: Concepts and Techniques, Second Edition, *Morgan Kaufmann Publishers*, 2006.

[6] R. Barandela, J.S. Sanchez, V. Garc´a and E. Rangel, Strategies for learning in class imbalance problems, *Pattern Recognition*, pp. 849–851, 2003.

[7] J. Derrac, S. Garcia, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, *Swarm and Evolutionary Computation*, Vol. 1, pp. 3–18, 2011.