

Efficient Associative Classification using Genetic Network Programming

S.P. Syed Ibrahim
Assistant Professor
Department of CSE
PSG College of Technology
Coimbatore

K.R.Chandran
Professor of IT and Head
Department of Computer and Information
Sciences
PSG College of Technology
Coimbatore

ABSTRACT

Classification and association rule mining are the two important tasks addressed in the data mining literature. Associative classification method applies association rule mining technique in classification and achieves higher classification accuracy. Associative classification method typically yields a large number of rules, from which a set of high quality rules are chosen to construct an efficient classifier. Hence generating a small subset of high-quality rules without jeopardizing the classification accuracy is of prime importance but indeed a challenging task. This paper proposes an efficient information gain based associative classification method using genetic network programming, which generates sufficient number of rules to construct the accurate classifier. Experimental results show that, the proposed method outperforms the existing genetic network based associative classification method and traditional decision tree classification algorithm.

General Terms

Data Mining, Information Extraction

Keywords

Evolutionary computation, data mining, Genetic network programming, Associative Classification.

1. INTRODUCTION

Data mining principally deals with extracting knowledge from dataset. In the world where data is all around us, the need of a hour is to extract knowledge or interesting information, which is hidden in the available data. Association rule mining is concerned with extracting a set of highly correlated features shared among a large number of records in a given database. It uses unsupervised learning where no class attribute is involved in finding the association rule. On the other hand, classification uses supervised learning where class attribute is involved in the construction of the classifier to predict the new instance. Both, association and classification are significant and efficient data mining techniques.

Associative classification is a recent and rewarding technique that applies the methodology of association into classification and achieves high classification accuracy. Associative classifier construction is of two steps. In the first step, all the class association rules are generated based on association rule mining technique. Then the rules are ranked and the rules that satisfy certain threshold conditions are used to construct the classifier. After rule ranking, only the high-ranking rules are chosen to build the classifier and the rest are pruned.

This paper proposes genetic network based associative classification, which constructs a genetic network based on information gain. This network will provide sufficient numbers of class association rules. Then support, confidence and correlation based measures are applied to the ruleset to filter the less important rules. Then the obtained optimal ruleset can be used for prediction.

The rest of the paper is organized as follows: Section 2 gives an insight about the past work in this field and section 3 explains the construction of genetic network. The last section presents the experimental results and observations followed by the conclusion.

2. RELATED WORK

Associative classification was first introduced by Liu et al (1998) which focus on integrating two known data mining tasks, association rule discovery and classification. The integration is focused on a special subset of association rules whose right hand side is restricted to the class attribute; for example, consider a rule $R: X \rightarrow Y$, Y must be a class label. Associative classification generally involves two stages. In the first stage, it adopts the association rule generation methods like Apriori candidate generation (Agarwal and Srikant, 1994), or FP growth (Han et al., 2000) algorithms to generate class association rules. For example CBA (Liu et al., 1998) method employs Apriori candidate generation and other associative methods such as CPAR (Yin and Han, 2003), CMAR (Li et al., 2001) while Lazy associative classification (Baralis et al., 2004, 2008) methods adopts FP growth algorithm for rule generation. The rule generation step generates huge number of rules. Experimental results reported by Baralis et al., (2008) has shown that the CBA method which follows Apriori association rule mining algorithm generates more than 80,000 rules for some datasets that leads to memory exceptions and other severe problems, such as overfitting etc.,

If all the generated rules are used in the classifier then the accuracy of the classifier would be high but the process of classification will be slow and time-consuming. So several rule pruning techniques are proposed to choose an optimal rule set. To apply rule pruning, the generated rules are ranked based on several parameters and interestingness measures such as confidence, support, lexicographical order of items etc. In CBA method, the rules are arranged based on their confidence value. If two rules have the same value for the confidence measure then the rules are sorted based on their support. If both confident and support values are same for two rules then the sorting is

done based on their rule length. Even after considering confidence, support, and cardinality measures, if there exists some rules with the same values for all three measures then the rules are sorted based on its lexicographic order as in Lazy pruning (Baralis et al., 2008) method.

After rule ranking, CBA method uses database coverage method to prune some rules, in order to construct an optimal rule set. Database coverage chooses the highest ranked rule and checks it against the training data set. If it covers at least one training data element, then, it will be considered for the construction of the classifier. This process is repeated until all the sorted rules or training objects are covered. One of the well known discrete data hypothesis testing is the Chi-square test (Snedecor and Cochran, 1989). This method evaluates the correlation between two items i.e. it checks whether the two items are positively correlated or negatively correlated. Only positively correlated rules are used in the classifier. This method is followed in the CMAR (Han and Pei, 2001). CPAR (Yin and Han, 2003) uses the Laplace accuracy measure to estimate the expected error rate for each rule. Then the best rules are selected to construct the classifier.

A recent approach for rule pruning is lazy pruning (Baralis et al., 2008) where a rule is pruned only if it misclassifies the data. The entire ruleset is segregated into three sets namely, useful rules, harmful rules and spare rules. A rule which classifies at least one data item correctly is said to be a useful rule and that which misclassifies a data item is a harmful rule and the leftovers are the spare rules which are not pruned but used when needed. Though lazy pruning strategy works well for small datasets, in the case of large datasets, there exist constraints in memory space and ruleset quality.

Chen et al. (2006, 2011) proposed a new approach based on information gain where more informative attributes are chosen for rule generation. An informative attribute centred rule generation produces a compact ruleset.

Syed et al., (2011a) proposed compact weighted associative classification based on information gain, where class association rule generation algorithm randomly chooses one non class attribute from dataset and all the items are generated only based on that attribute. Thus this algorithm reduces number of itemset generation. Finally the algorithm calculates the weighted support and weighted confidence for each item and determines whether the item is frequent or not.

Evolutionary based associative classification method (Syed et al., 2011b) was proposed recently. This approach takes subset of rules randomly to construct the classifier. Richness of the ruleset was improved over the generation. Syed et al. (2011c) proposed statistical based rule ranking method. Here, after generating the rules using associative classification rule generation algorithm, rules are ranked based on statistical measure.

Eguchi et al., (2006) proposed Genetic Network Programming (GNP) Based evolutionary approach for Multi-agent Systems.

GNP consists of the directed graph type gene structure where helps in analyzing the decision making mechanism of agents.

Shimada et al., (2006) used GNP construction method to construct the associative classifier, which constructs the GNP in random and extracts class association rules directly from the GNP. Random GNP construction generates huge number of rules that leads to memory exception and consumes more CPU.

To improve the performance of GNP based associative classification this paper proposes information gain based GNP construction.

3. THE PROPOSED METHOD

3.1 Genetic network construction

Genetic Network Programming (GNP) introduced by Eguchi et al., (2006) is one of the evolutionary optimization techniques, which uses the directed graph structures to solve the problems. The GNP is proposed to overcome the problem of Genetic Programming. In Genetic programming searching for an optimum solution is very difficult because of its bloated space but the basic structure of GNP genome structure is networks. So this can reduce the searching space for the solutions that leads to good performance than GP.

GNP consists of three nodes: Start node, Judgment node and Processing node. Start node defines the start of the genetic tree.

Judgment node and processing node work as decision making and process function respectively. Judgment nodes are the set of J_1, J_2, \dots, J_n , and Processing nodes are the set of P_1, P_2, \dots, P_n . Here attributes are behaved as Judgment node and Class values behave as the processing node.

3.2 Attribute selection based on information gain

Information gain is a measure that is used in information theory to quantify the 'information content' of messages (Han and Kamber, 2001). In ID3 decision tree algorithm (Quinlan, 1986) information gain is used to choose the best split attribute.

In this paper, information gain is used to construct the initial genetic tree network. In the process of constructing genetic tree network, information gain is used to select the best attribute instead of considering all the nodes as in random tree. By doing this, only those attributes with maximum information gain are selected for rule generation. Figure 1 shows the work flow of proposed system. Information gain measure is used to identify the best split attribute in decision tree classifier. In this paper, it could be used to construct the efficient genetic network. In the process of constructing the genetic network, instead of considering all the nodes, information gain measure will be used to select the best splitting attribute. In this way, genetic network constructs the decision tree based genetic network.

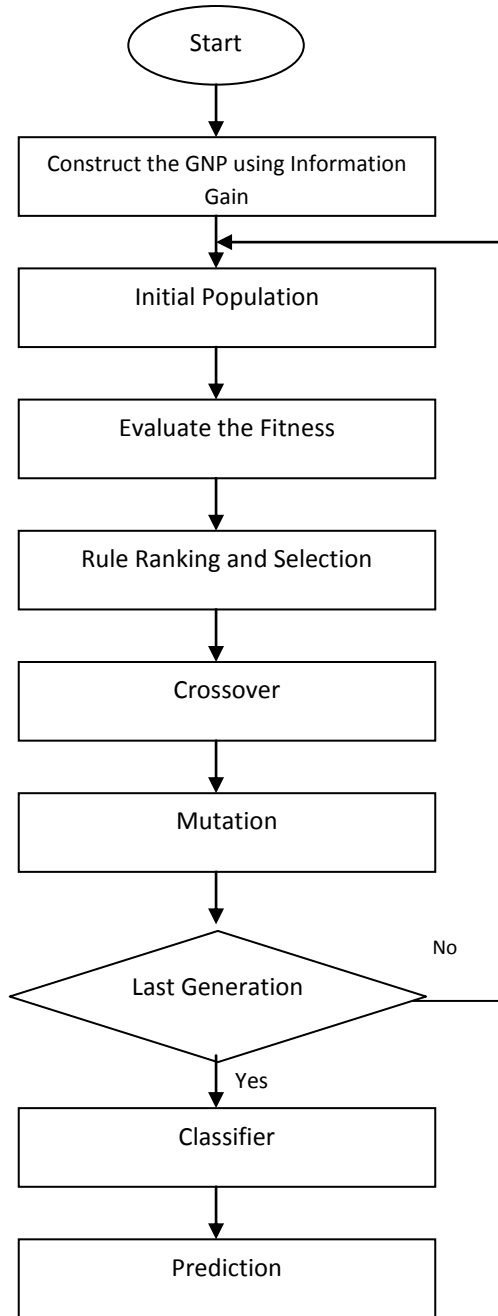


Fig 1: The Proposed System

Suppose an attribute A has n distinct values that partition the dataset D into subsets T1, T2, ..., Tn. For a dataset, $\text{freq}(C_k, D) / |D|$ represents the probability that a tuple in D belongs to class C_k .

Then $\text{info}(D)$ is defined as follows to measure the average amount of information needed to identify the class of a transaction in D:

$$\text{Info}(D) = - \sum_{k=1}^g \frac{\text{freq}(C_k, D)}{|D|} \times \log_2 \frac{\text{freq}(C_k, D)}{|D|} \quad (1)$$

Where $|D|$ is the number of transactions in database D and g is the number of classes. After the dataset D is partitioned into n values of attribute A, the expected information requirement could be defined as:

$$\text{info}_A(D) = \sum \frac{|D_i|}{|D|} \times \text{info}(D_i) \quad (2)$$

The information gained by partitioning D according to attribute A is defined as:

$$\text{Gain}(A) = \text{info}(D) - \text{info}_A(D) \quad (3)$$

The best split attribute is the one that maximized the information gain in the data set D. These best attributes are used to construct the genetic network. For example consider the Table 1, the CD4 Cell Count attribute has highest information gain. So this attribute will act as root judgment node and it has three different decision branches.

Table 1. Sample Dataset

CD4 Cell Count	Sweating at Night	Tuberculosis (TB)	Temperature	AIDS
>500	High	no	Normal	no
>500	High	no	High	no
<200	High	no	Normal	yes
200 .. 500	Medium	no	Normal	yes
200 .. 500	Nil	yes	Normal	yes
200 .. 500	Nil	yes	High	no
<200	Nil	yes	High	yes
>500	Medium	no	Normal	no
>500	Nil	yes	Normal	yes
200 .. 500	Medium	yes	Normal	yes
>500	Medium	yes	High	yes
<200	Medium	no	High	yes
<200	High	yes	Normal	yes
200 .. 500	Medium	no	High	no

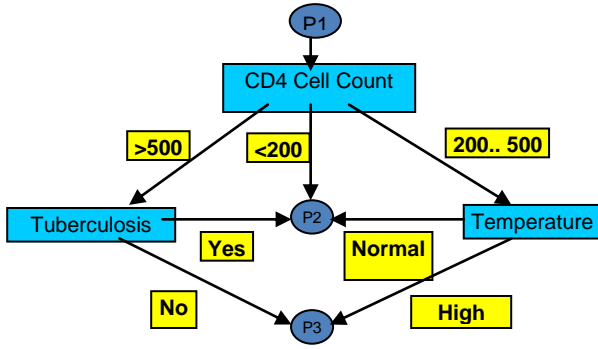


Fig 2: GNP Structure for Sample Dataset

Then again information gain is applied to the remaining dataset and identifies the other judgement nodes. Finally class attribute values are behave as processing node. Figure 2 shows the GNP structure for the sample dataset given in the Table 1.

3.3 Initial population

After constructing the GNP, class association rules can be extracted directly from the GNP. And these initial class association rules are known as initial population.

For example from GNP structure, the following class association rules can be generated.

{CD4 Cell Count > 500, Tuberculosis (TB) = Yes} → {AIDS = yes}

{CD4 Cell Count > 500, Tuberculosis (TB) = No} → {AIDS =no}

{CD4 Cell Count < 200} → { AIDS = yes }

{CD4 Cell Count 200 ..500, Temperature = Normal}→ {AIDS = yes }

{CD4 Cell Count 200..500, Temperature = High}→ {AIDS =no}

These generated rules are stored in the initial poll. Here rule fitness evaluation methods are applied to sort the rules in an efficient manner.

3.4 Fitness evaluation

Fitness is a goodness measure of a chromosome. The proposed approach uses two different measures to measure the fitness of the each rule. They are 1) Lift evaluation method and 2) Chi square fitness evolution.

3.4.1 Lift measure

Lift is the one of the interestingness measure. It is defined as follows:

$$Lift(X, Y) = P(XUY) / P(X)P(Y) \quad (4)$$

where, $P(X \cup Y)$ is the probability of the occurrence of itemset X and itemset Y. $P(X)$ is the probability of the occurrence of itemset X, and $P(Y)$ is the probability of the occurrence of itemset Y.

This Lift measure was introduced to overcome the disadvantages of confidence measure. Confidence measure does not take the frequency of the consequent into consideration. In the lift measure $P(Y)$ is appeared in the denominator. So lift can also be defined as confidence divided by the frequency of Y.

The lift measure is defined over $[0, \infty]$ and its interpretation is as follows:

- If lift < 1, then X and Y are said to be negatively correlated.
- If lift = 1, then X and Y are said to be independent of each other.
- If lift > 1, then X and Y are said to be positively correlated.

Fitness of each individual with Lift measure is defined by

$$F = \sum (Lift(i) + (No.of. antecent(i) - 1) + (No.of. Consequent(i) - 1)) \quad (5)$$

3.4.2 Chi square fitness evaluation

The Chi-square test is one of the well known statistical testing methods. This method compares the observed data with the expected data according to a specific hypothesis.

Fitness of GNP with chi square measures was defined as in (Shimada et al., 2006).

$$F = \sum (X^2(i) + (No.of antecent(i) - 1) + (No.of. Consequent(i) - 1)) \quad (6)$$

If $X \rightarrow Y$, X is antecedent, y is consequent. Here Y must be the class.

$$Support(X \rightarrow Y) = P(XUY) / N \quad (7)$$

$$Confidence(X \rightarrow Y) = \frac{P(XUY)}{P(X)} \quad (8)$$

Calculation of χ^2 value of the rule $X \rightarrow Y$ is described as follows. Let support (X) = x, support (Y) = c, support(X U Y) = z and the number of transactions in the dataset equal N.

Table 2 shows the contingency of X and C: the upper parts are the expectation values under the assumption of their independence, and the lower parts are observational. Now, let E denote the value of the expectation under the assumption of independence and O the value of the observation. Then the χ^2 statistic is defined as follows:

Table 2. Chi Square Computation

	C	¬C	Σmax
X	Nxc Nz	N(x-xc) N(x-z)	Nx
¬X	N(c-xc) N(c-z)	N(1-x-c+xc) N(1-x-c+z)	N(1-x)
Σ	Nc	N(1-c)	N

$$\chi^2 = \frac{(Observed - Expected)^2}{Expected} \quad (9)$$

$$Chisquare \chi^2 = \frac{N(z - xc)^2}{xc(1-x)(1-c)} \quad (10)$$

This has 1 degree of freedom. If it is higher than a cutoff value 3.84 at the 95% significance level, then the rule is accepted or the rule is rejected. After rule evaluation, the rules having highest fitness will be separated and stored in a poll. To apply rule pruning, the generated rules are ranked based on several parameters and interestingness measures such as confidence, support, lexicographical order of items etc.

3.5 Cross Over

Crossover or the recombination operator forms offspring(s) or new individuals by combining certain portions of two individuals (parents) currently in the population. There are various types of recombination methods such as single point crossover, two point crossover, order based crossover, cycle crossover etc. This proposed method single point crossover.

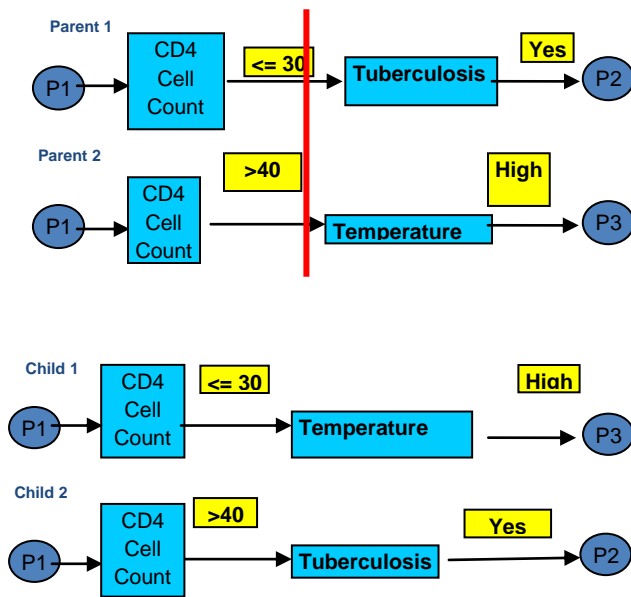


Fig 3: Crossover Operation

The steps involved in a single point crossover are as follows:

1. Two parents are selected.
2. A crossover point is chosen at random.
3. Offspring 1 is built by combing the left side genes of the parent 1 with the right side genes of the parent 2 taking crossover point as the median.

Offspring 2 is built by combing the right side genes of the parent 1 with the left side genes of the parent 2 taking crossover point as the median. Figure 3 shows the cross over operation.

3.6 Mutation

Mutation is a variation operator that induces a change in the genotype of the individual which in-turn gets reflected in the phenotype. The most common method way of implementing mutation is to select a bit at random and flip (change) its value.

Here 2 different types of mutation is introduced. 1) Mutating the Judgement node. 2) Mutating the value of the judgement node.

3.6.1 Mutation I: Mutating the Judgement node

In this type of mutation, any one of the judgement node is randomly selected and replaced with another judgement node. For example consider the Figure 4, the Temperature judgment node and its value is changed with another judgement node Tuberculosis. This type of mutation is same as cross over. Figure 5 shows the result of mutation I.

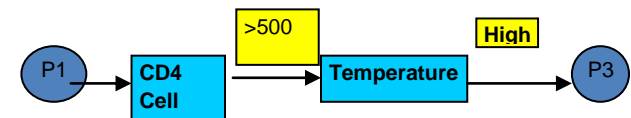


Fig 4: Before Mutation I

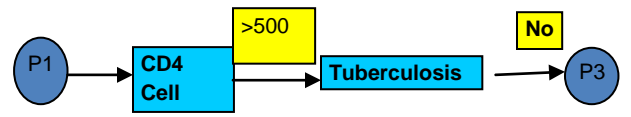


Fig 5: After Mutation I

3.6.2 Mutation II: Mutating the value of Judgement Node

In this type of mutation, judgement node is randomly selected and its value is replaced with any one of the attribute value

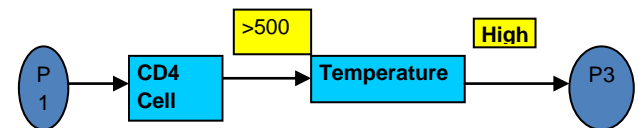


Fig 6: Before Mutation II

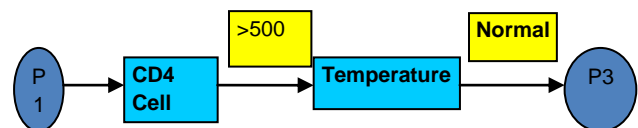


Fig 7: After Mutation II

For example consider the Figure 6 and 7, the value of temperature judgement node is changed from High to Normal. This change will produce new off spring.

4. EXPERIMENTAL RESULTS

The proposed system was tested using benchmark datasets from the University of California at Irvine Repository (UCI). The datasets were pre-processed to convert to a general format. A brief description about the datasets was presented in Table 3.

The experiments were carried out on a PC with Intel Core 2 Duo CPU with a clock rate of 1.60 Ghz and 2 GB of main memory. Holdout approach (Han and Kamber, 2008) was used where 90% of the data were randomly chosen from the dataset and used as training dataset and remaining 10% were used as the testing dataset. The training dataset is used to construct a model for classification. After constructing the classifier, the test dataset is used to estimate the classifier performance.

Table 3. Dataset Descriptions

Dataset	Transactions	Classes	Items
Breast	699	2	18
Ecoli	336	8	34
Glass	214	7	20
Hepatitis	155	2	33
Pima	768	2	36
Zoo	101	7	34

4.1 Accuracy Computation

Accuracy measures the ability of the classifier to correctly classify unlabeled data. It is the ratio of the number of correctly classified data over the total number of given transactions in the test dataset. Construction of the classifier and classification of the test dataset is done ten times and the overall accuracy is obtained by calculating the average of the accuracy values obtained from the ten different runs. The performance of the proposed algorithm is evaluated by comparing it with the traditional decision tree classification algorithm and GNP based class association rule mining algorithm proposed by shimada et al.,(2006). Table 4 shows the accuracies for the different datasets using chi square measure.

The authors chen et al., (2006) shows setting the confidence value greater than 50 eliminates the conflicting rules. So here confidence value is set greater than 50. The proposed method with mutation I has about +4.11 percent improvement against the traditional decision tree classifier and about +8.08 percent over the existing system and with mutation II, it has about +4.51 percent and +8.94 percent improvement against the traditional decision tree classifier and the existing system respectively. So the proposed system performs better than the traditional classifier and the existing genetic network based associative classifier.

Table 5 shows the accuracies for the different datasets using lift measure. The proposed method with mutation I has about +4.05 percent improvement against the traditional decision tree classifier and about +9.41 percent over the existing system and with mutation II, it has about +4.34 percent and +6.84 percent improvement against the traditional decision tree classifier and the existing system respectively.

Table 4. Accuracy Comparison Using Chi-Square Measure

Dataset	Decision Tree	GNP-Random (chi >3.84 Conf >50 Sup >1)	GNP-IG (chi >3.84 Conf >50 Sup >1)	GNP-Random (chi >3.84 Conf >50 Sup >1)	GNP-IG (chi >3.84 Conf >50 Sup >1)
		Mutation I		Mutation II	
Breast	94.99	93.44	98.14	94.70	98.14
Ecoli	81.54	75.75	75.14	73.32	75.14
Glass	73.83	77.56	79.04	78.57	81.42
Hepatitis	82.58	80.66	91.99	80.68	91.99
Pima	78.25	72.07	87.65	74.41	87.65
Zoo	92.07	80.00	96.00	75.00	96.00
Average	83.88	79.91	87.99	79.45	88.39

Table 5. Accuracy Comparison Using Lift Measure

Dataset	Decision Tree	GNP-Random (lift >1.00 Conf >50 Sup >1)	GNP-IG (lift >1.00 Conf >50 Sup >1)	GNP-Random (lift >1.00 Conf >50 Sup >1)	GNP-IG (lift >1.00 Conf >50 Sup >1)
		Experiments with Mutation I		Experiments with Mutation II	
Breast	94.99	95.71	98.14	95.71	98.28
Ecoli	81.54	75.45	73.63	75.45	73.24
Glass	73.83	76.40	81.90	76.40	82.85
Hepatitis	82.58	80.66	91.99	80.66	91.99
Pima	78.25	74.93	87.92	74.93	88.95
Zoo	92.07	80.00	94.00	85.14	94.00
Average	83.88	78.52	87.93	81.38	88.22

Table 6. Running Time Comparisons

Dataset	GNP- Random with chi square	GNP- IG with chi square TIME Sec	GNP- IG with lift TIME Sec	GNP- IG with chi square	GNP- IG with lift
	TIME Sec	Experiments with Mutation I		Experiments with Mutation II	
Breast	2344	698	712	686	699
Ecoli	584	254	253	249	255
Glass	596	299	297	302	298
Hepatitis	749	138	139	120	131
Pima	2182	383	396	366	373
Zoo	244	129	140	133	141
Average	1116.50	316.83	322.83	309.33	316.17

5. CONCLUSION

The main objective of this paper is to introduce a new associative classification method based on genetic network programming. The proposed GNP based associative Classification constructs the genetic network based on information gain, which generates sufficient number of high quality class association rule. It is interesting to further enhance this proposed method to reduce the CPU time and cost by reducing number of attributes. The experiments are done on several datasets which validates the proposed method. The experimental results show that the proposed GNP based on Information gain method outperformed the traditional classifier and genetic network based associative classifier.

6. REFERENCES

[1] Agarwal.R and Srikant.R, (1994). "Fast algorithm for mining association rules in large data bases", in the proceedings of the 20th international conference on very Large Data Base (VLDB '94), Santiago, Chile, pp 487-499.

[2] Baralis, E. and Torino, P (2004). "A Lazy approach to pruning classification rules" in the proceedings of the IEEE International Conference on Data Mining (ICDM'02), Maebashi City, Japan, pp 35-42.

[3] Baralis, E., Chiusano, S. and Graza, P (2008). "A lazy approach to associative classification". *IEEE Transactions on Knowledge and Data Engineering*, VOL. 20, NO. 2, February, pp 156 – 171.

[4] Baralis, E., Chiusano, S. and Graza, P (2004). "On support thresholds in associative classification". In proceedings of the 2004 ACM Symposium on Applied Computing. Nicosia, Cyprus: ACM Press.

[5] Chen. G, Liu. H, Yu. L, Wei.Q and Zhang.X (2006), "A new approach to classification based on association rule mining", *Decision Support Systems*. pp- 674– 689.

[6] Eguchi. T, Hirasawa. K, Hu. J, and Ota. N (2006) "A study of Evolutionary Multiagent Models Based on Symbiosis," *IEEE Transaction on System, Man and Cybernetics*, vol.36, no.1, pp. 179-193,

[7] Han. J, Pei.J, and Yin. Y (2000), "Mining Frequent Patterns without Candidate Generation," *Proceedings of ACM SIGMOD*.

[8] Han.J and Kamber.M (2001), "Data Mining: Concepts and Techniques". New York: *Morgan Kaufmann Publishers*.

[9] Hirasawa. K, Okubo. M, Katagiri. H, Hu. J, and Murata. J (2001), "Comparison between Genetic Network Programming (GNP) and Genetic Programming (GP)" in *Proceedings of IEEE Congress on Evolutionary Computation*, pp. 1276- 1282.

[10] Li, W., Han, J. and Pei, J, (2001). "CMAR: Accurate and efficient classification based on multiple-class association rule". In *Proceedings of the International Conference on Data Mining (ICDM'01)*, San Jose,CA, pp. 369–376.

[11] Liu, B., Hsu, W. and Ma, Y (1998) "Integrating classification and association rule mining". In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*. New York, NY: AAAI Press, pp. 80–86.

[12] Merschmann. L, Plastino. A (2007), "A lazy data mining approach for protein classification". *IEEE Transactions on Nanobioscience*, vol. 6, no. 1, Pp. 36-42.

[13] Merschmann. L, Plastino. A (2010) "HiSP-GC: A Classification Method Based on Probabilistic Analysis of Patterns" *Journal of Information and Data Management*, Vol. 1, No. 3, October, Pages 423–438.

[14] Quinlan. J (1986). "Induction of decision trees". *Machine Learning*, pp, 81–106.

[15] Shimada. K, Hirasawa. K, and Hu. J (2006) "Class Association Rule Mining with Chi-Squared Test Using Genetic Network Programming" in the proceedings of 2006 *IEEE International Conference on Systems, Man, and Cybernetics*, Taipei, Taiwan. October 8-11, pp. 5338 – 5344.

[16] Snedecor.W, and Cochran. W (1989), "Statistical Methods", *Eighth Edition, Iowa State University Press*.

[17] Syed Ibrahim. S.P, Chandran.K.R, Abinaya.M.S (2011a), "Compact Weighted Associative Classification" in the *IEEE International Conference on Recent Trends in Information Technology (ICRTIT 2011)*, MIT, Anna University.Chennai, June 3-5, 2011

[18] Syed Ibrahim.S.P, Chandran.K.R, Jabez Christopher.J (2011b), "An Evolutionary Approach for Ruleset Selection in a Class Based Associative Classifier" in the *European*

Journal of Scientific Research, ISSN 1450-216X, Vol.50
No.3, pp.422-429.

- [19] Syed Ibrahim.S.P, Chandran.K.R, Muniasamy.R (2011c), “Efficient Rule Ranking And Rule Pruning In Associative Classification” in *the Conference On Research Issues In Engineering And Technology (Computer Science And Engineering Stream)*, Organized by PSG College of Technology, Coimbatore, India, April 28.
- [20] Syed Ibrahim. S.P., K.R.Chandran, R.V.Nataraj (2011d), “LLAC: Lazy Learning in Associative Classification” in *the Springer Lecture Series in Communications in Computer and Information Science (CCIS), Advances in Communication and Computers*, 2011, 190, Part I, PP. 631 – 638.
- [21] Yin.X and Han.J (2003) , “CPAR: Classification Based on Predictive Association Rules,” in *the proceedings of Third SIAM International Conference on Data Mining (SDM’03)*.
- [22] Zhang. X, Chen.G, Wei. Q (2011), “Building a highly-compact and accurate associative classifier” *Applied Intelligence*, Volume 34, Number 1, PP.74-86.

7. AUTHORS PROFILE

S. P. Syed Ibrahim received his BE in 2004 from Bharathidasan university and ME in 2006 from Anna University, Chennai. He has been research scholar of Anna Univerisy – Coimbatore. He is currently as Assistant professor in the department of Computer Science and Engineering at PSG College of Technology, Coimbatore. He has published more than 15 research papers in various journals and conferences. His research interest includes data mining, storage management.

K.R. Chandran received his BE in 1982 and ME in 1986 from PSG College of Technology and PhD in 2006 from Bharathiar university, Coimbatore. He is currently professor in the department of Information Technology at PSG College of Technology, Coimbatore. He is a Fellow member in Institution of Engineers (India) and life member of ISTE, ACS, and The Textile Association (INDIA). He has published more than 55research papers in various journals and conferences. His research interest includes software engineering, computer networks, data mining, system modeling and simulation.