Classifier Learning Algorithm Based on Genetic Algorithms

Li-yan DONG[1] Guang-yuan LIU[1] Sen-miao YUAN[1] Yong-li LI[1,2] Zhen Li[1]
1.College of Computer Science and Technology, Jilin University, Changchun,130012, P.R.China
2.College of Computer, Northeast Normal University Changchun,130024,P.R.China
Email: dongliyan@gmail.com

Abstract

The paper addresses the problem of classification. A restricted BAN classifier learning algorithm－GBAN based on genetic algorithm is proposed. Genetic algorithm is used in this new algorithm to study the network structure, this can reduce complexity of calculation substantially. Meanwhile, the network structure of TAN classifier is extended by restricting the complexity of the structure of BAN classifier., and then a restricted BAN classifier is obtained. To learn the structure of this kind classifier, fitness function based on logarithm likelihood and the corresponding genetic operator are designed, network structure code scheme is also designed. As a result, this algorithm can converges on the overall optimal structure. Experimental result shows that GBAN algorithm performs better than TAN algorithm and has a better accuracy when the relationship between attributes of a data set is relatively complicated.

Key Words: Bayesian network classifier, Genetic algorithm, logarithm likelihood

1. Introduction

Naïve Bayesian Classifier is one of the simple and efficient probabilistic ways of classifying, which shows very good result under the assumption of "class conditional independence"[1][2], that is, given the class label of a tuple, the values of the attributes are assumed to be conditionally independent of one another. Unlike other ways of classifying, this assumption simplifies computation. To get parameter of each attribute, only the values' frequencies of each attribute in the training set need to be calculated and there is no need for searching. But, in the real world, it's hard to meet the assumption of conditional independence, so many people are trying to find some other models that can work efficiently without this assumption. TAN classifier(Tree Augmented Naïve Bayes)[3] can obtain MWST(maximum weight span tree) based on Chow-Liu' MWST algorithm, which loosen the assumption of class conditional independence while keep the structure simple and efficient. But the accuracy of TAN classifier decreases when relationships between attributes of data set become more complex. BAN classifier (Bayes Network Augmented Naïve-Bayes)[4] loosen the assumption further, and shows stronger ability in learning when the relationships between attributes is complex. In nature, the process of BAN learning is a process of constrained learning process of Bayesian Networks which requires searching in large space and complex computation, and may easily trap in the problem of local optimal solution. In this paper, we propose a improved TAN algorithm- GBAN (genetic algorithm base BAN) classification algorithm which is a combination of Bayesian Theory and the merit of genetic algorithm in searching.

2. Classifier Based on GBAN Learning Algorithm

Definition 1.GBAN algorithm is a BAN classifier learning algorithm based on genetic algorithm.

We confine networks learnt by GBAN to the following constraint: Given $A_1,...,A_n$ are attribute nodes, C is classification node, the network structure learnt by GBAN must meet:
(1)C is a parent of all the other node;
(2)All Attribute nodes form a Bayesian Networks.
(3)For a attribute node A_i , i=1,…,n, it can has at most m parents nodes except C.(Usually, m $\leq$ 4[5],[6]).

From above presentation, we can see GBAN classifier is a constrained BAN classifier or a extended TAN classifier. When m=1 and attributes nodes $A_1,\dots,A_n$ form a maximum weight span tree, we can get a TAN classifier.

In nature, the process of GBAN learning is a kind of learning Bayesian Networks which include learning structure and learning parameter. Since we can get parameters from the structure and data set easily, it's most important to learn the structure in Bayesian Network Learning. Given a data set D, we need to find out a Bayesian network that can match D most. A commonly used way is defining a scoring function, which can reflect the degree of matching between independence relationship showed by the structure and the sample data, then we use proper a search algorithm to find a network that get highest score.

Given variable set X, there are n!·2n possible Bayesian Networks. So the search space is very large. It's a NP problem to search every possible network and there may exist many network structures which has local best result. Genetic Algorithm can get optimal solution approximately in complex space, so it is suitable for solving the problem of learning Bayesian network and

0-7695-2882-1/07 $25.00 ©2007 IEEE

for learning BAN classifier.

3. GBAN Algorithm

Genetic algorithms attempt incorporate ideas of natural evolution, and it simulate the process of evolution in the nature. It uses fitness function as evaluation criterion, and by the operations of selection, crossover, mutation and so on among individuals of population, realizes the reformation of individual structure through iterative operating process. GBAN algorithm is an improved BAN algorithm based on Genetic algorithm.

3.1 Fitness Function

Definition 2. The GBAN algorithm use formula(3.1) as fitness function

$$\sum_{i,\pi(i)>0} I_{\hat{P}_D}(A_i;A_{\pi(i)} \mid C) \qquad (3.1)$$

where $\hat{P}_D()$ is experiential distributing of incident D, that is $\hat{P}_D(A)=\frac{1}{N}\sum_j l_A(u_j)$,and for incident $A \subseteq Val(U)$,when $u \in A$, $l_A(u)=1$;when $u \notin A, l_A(u)=0$.

A_i is attribute node, $A_{\pi(i)}$ represents the parents node set of Ai except C.

It's proven by Frideman that the nonspecialized scoring function we usually used，such as MDL[7] and Bayesian scoring function is not fit for representing a good Bayesian classifier. The reason for TAN has a high accuracy and efficient is that the networks learn by TAN can maximize the logarithm likelihood function. As an extension of TAN classifier, the GBAN classifier can use a extension of logarithm likelihood function as a criterion for judge whether the network is good, that is we use it as a fitness function and our aim is to maximize it. formula(3.1) is a fitness function for GBAN algorithm.

Definition 3. Let D be data set with a size of N ,which include attribute $C,A_1,\dots,A_n$, Network structure B_S is learned from data set D, logarithm likelihood can be defined as

$$LL(B_S \mid D)=\sum_{i=1}^{N}\log(P_B(u_i)) \qquad (3.2)$$

We can decompose $LL(B_S|D)$ into 3.3

$$LL(B_S \mid D)=N\cdot\sum_{X_i} I_{\hat{P}}(X_i;\prod\nolimits_{X_i}) + C \qquad (3.3)$$

(where $C$is a constant.)

So, maximizing $LL(B_S|D)$, we just need to maximize $\sum_{X_i} I_{\hat{P}}(X_i;\prod_{X_i})$.

Definition 4. The network structure by GBAN learning can be define by π (•):

(1) Equation $\pi:\{1,\dots,n\}|\!\rightarrow\{0,\ \dots,m\}$ is projection from $\{1,\dots,n\}$ to $\{0,\dots,m\}$ on node set $\{C,A_1,\dots,A_N\}$;

(2) π (i) is the number of Ai'parants(except classification node C).We let π(C)=0。

(3) There exist no sequence ,such that π $(i_j)=i_{j+1}$， i⩽j⩽ k， π $(i_k)=i_1$, i.e, there is no circle in the network.

Because classification node C do not have parent, so $I_{\hat{P}_D}(C;\prod_C)=0$

According the definition above, we can get:

$$I_{\hat{P}_D}(A_i;\prod\nolimits_{A_i})=\begin{cases} I_{\hat{P}_D}=(A_i;A_{\pi(i)},C)\dots if\pi(i)>0 \\ I_{\hat{P}_D}(A_i;C)\dots\dots\dots\dots otherwise \end{cases}$$

so, in the structure by GBAN, $\sum_{X_i} I_{\hat{P}_D}(X_i;\prod_{X_i})$ can be expressed as

$$\sum_{i,\pi(i)>0} I_{\hat{P}_D}(A_i;A_{\pi(i)},C)+\sum_{i,\pi(i)=0} I_{\hat{P}_D}(A_i;C) \qquad (3.4)$$

and according to the chain rule of mutual information：

$$I_P(X;Y,Z)=I_P(X;Z)+I_P(X;Y \mid Z)$$

From formula 3.4,we can induce formula 3.5

$$\sum_{i,\pi(i)>0} I_{\hat{P}_D}(A_i;A_{\pi(i)} \mid C)+\sum_i I_{\hat{P}_D}(A_i;C) \qquad (3.5)$$

Because the value of $\sum_i I_{\hat{P}_D}(A_i;C)$ has nothing to do with parent nodes π (i) of each node, that is, has nothing to do with network structure, we can maximize $LL(B_S|D)$ if our network structure can maximize the first term $\sum_{i,\pi(i)>0} I_{\hat{P}_D}(A_i;A_{\pi(i)} \mid C)$.

Since GBAN algorithm take advantage of genetic algorithm, and use the first term $\sum_i I_{\hat{P}_D}(A_i;C)$ as fitness function, according to constringency of genetic algorithm, GBAN can converge to the structure, which GBAN limits to and can maximize logarithm likelihood, with probability nearly to 1.

3.2. Encoding Method

In GBAN algorithm, individual is corresponding the network structure S in the Bayesian classifier. We encode network structure by adjacency matrix C=(cij), where i,j=1,…,n. We do it as follows:

$$C_{ij}=\begin{cases}1, & \text{if there is edge } i\rightarrow j; \\ 0, & otherwise\end{cases} \qquad (3.6)$$

Since classification node is the parent node of all the other attribute nodes, and this characteristic will not be changed in the process of evolution, that is, it does not participate heredity operations, we don't need to code to the classification node. With this encoding method , a individual in genetic algorithm can be denoted by a string of 0/1:

$$C_{11}C_{21}\dots C_{n1}C_{12}C_{22}\dots C_{2}\dots C_{1n}C_{2n}\dots C_{nn}$$

Adjacency matrix (denoted by a string) can be viewed as a chromosome， where each row is a gene and

$c_{1i}c_{2i},\ldots,c_{ni}$ is allele.

3.3. Genetic Opertaion

3.3.1. selection

We select individuals by a way called rank selection method[8],which put individuals into different ranks according to there fitness, and different rank has a different probability to be selected. As a result, the probability to be selected is associated to the individual's rank and has nothing to do with the absolute value of fitness,in this way, we can avoid a superior individual has too big probability to be selected and matures too early.

Let S_j^t denotes the j'th individual of the t'th generation, $rank(F(S_j^t))$ denotes the fitness rank of S_j^t, and then the probability $p_{j,t}$ that S_j^t to be selected is (λ is the scale of population):

$$p_{j,t} = \frac{rank(F(S_j^t))}{\lambda(\lambda+1)/2} \qquad (3.7)$$

3.3.2. crossover

The individuals selected by selection operator are paired randomly, and performed crossover operator in the crossover probability p_c . This article uses the common single point crossover method.

3.3.3. Mutation

In mutation, character of individual changes randomly in order to avoid local extremum. In this process, some value of genes are changed with a probability of p, where p is close to zero. In this paper, we adopt even mutation method.

After the process of crossover and mutation, there may exist some illegal individuals, we can classify these illegal individuals into two kind, one kind is that a individual is not a directed acyclic graph (DAG), another kind is some nodes (attribute node) have more than m parents node (except classification node). So ,we need an operation called repair.

3.3.4. Repair

For individuals that are not directed acyclic graph (DAG), we need to check if a newly produced individual is a DAG ,if not, just cancel the genetic operation done prviously.

It costs a lot to find a best or near best sub-set of edges in the adjacency matrix that make sure an individual is a DAG, and if we do all genetic operation like this, the genetic algorthm will be inefficient; meanwhile, individuals not removed from the population can retain the local outstanding structure, which can be passed to the individuals in the next generation.

For these nodes(attribute nodes) that have more than m parents in the individual, choose the best m parents in the parents-set, remove edges from others parents.

3.4 Setting of Initial Population

We initiate the population randomly, and confine initial population to the limitation of DAG and number of parents node as we discussed in 3.3

The number of population λ can not too large since it's complex to calculate the fitness. On the other hand, the number of population can not too small neither in order to avoid convergence when not mature. So the number of population is about 10~100. The stop condition can be chosen as such like, there more than 10000 generations or like the best structure of networks have not changed for g generations (where g is a constant, and usually g=20). The number of parents m can be chosen according to the number of attributes in database and the number of data, but all along m<=4.

3.5 GBAN algorithm

```
PROCEDURE GBAN     /* λ  is initial populate， t indicate the tth generation*/
BEGIN
1 Initiate population pop(0) randomly.
2 For the individual in pop(0)who has more than m number of parents or individual $I_0^i$ who is not a DAG , repair it according we discussed in section 3.3
3 Calculate the fitness for all individuals in pop(0).t=0
4 WHILE NOT stop_condition DO
    BEGIN
5       Do the choose operations. Choose parent(t) that to be used to generate the next generation。
6       Do crossover operations between parents, and then do the repair operations
7       For the new individuals generated in step 6,do the mutation and repair operations.
8       Generate pop(t+1)by add the new individuals generated by step 6,7 to pop(t),and reduce the size of pop(t+1) to λ according to elitist reduction criterion.
9       t＝t+1，keep the best individuals in the t th generation.
    END
10 Output the best individuals。
END
```

4 Experimental Results

We choose 11 data set from UCI Database[9] as our experimental data set, and preprocessed those data by using the preprocess tool PreProcessor[10] offered by Jie Cheng. We use ratio of inaccuracy as an indicator to judge classifier's performance. For large data set,, we use hold out(1/3) way, and for small data set, use 5-fold way to evaluate the modal respectively. According the way above ,we compared the performance of GBAN algorithm , Naïve Bayesian classifier and TAN classifier. The experimental result is as follows:

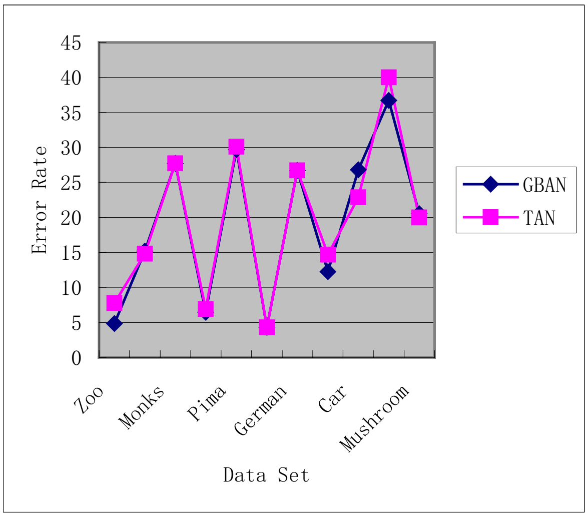

Figure 1. The ratio of inaccuracy showed by GBAN and TAN algorithm

From the mistake curve in figure 1. we can see that, for data sets which have many attribute, such as Zoo, Vote, Segment, Chess, GBAN classifier showed a better performance than TAN. While in other data set, the performance is about the same. We also see that the distance between points is not too far, so we can say the performance between them on the same data set is about the same.

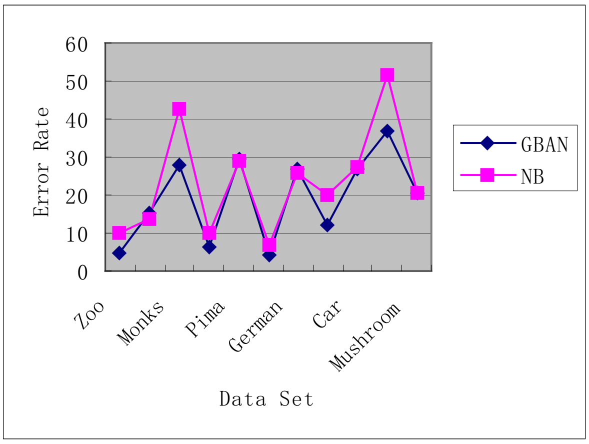

Figure 2. The ratio of inaccuracy showed by GBAN and NB

From the mistake curve in figure 2. we can see that, for most data sets, GBAN has a better performance than NB in classification, and distance between them in some point is far., this indicate, for some data set, especially those set data set whose attribute variables have strong relation with each other, GBAN performs better than NB in classification.

5 Conclusion

This paper addresses the problem of GBAN algorithm which learning constrained BAN classifier based on Genetic Algorithm. Experimental result shows, GBAN classifier shows a better performance and TAN classifier when the relationship between attributes is complex. Additional research will optimize the calculation of fitness function so that the efficiency will be better.

References

[1] R. Bouckaert. Naive bayes classifiers that perform well with continuous variables[C] // Proc Seventeenth Australian Joint Conference on Artificial Intelligence (AI 2004), Advances in Artificial Intelligence. Cairns, Australia: Springer, 2004:1089-1094,

[2] Remco R. Bouckaert. Bayesian Network Classifiers in Weka. http://citeseer.ist. psu.edu/705669.html

[3] Friedman N, Geiger D, Goldszmidt M. Bayesian Network Classifiers[J]. Machine Learning, 1997, 29(2-3): 131-161

[4]Cheng J,Greiner R.Comparing Bayesian network classifiers[C].//Proc.of the 15th Conf.on Uncertainty in Artificial Intelligence. San Francisco: Morgan Kaufmann Publishers,1999:101-108.

[5] Larran'aga P, Poza M, Yurramendi Y et al. Structure learning of Bayesian networks by genetic algorithms： A performance analysis of control parameters[J]. IEEE Trans on Pattern Analysis and Machine Intelligence,1996,18(9) ： 912~925

[6] LIU Da You, WANG Fei et al., Research on Learning Bayesian Network Structure Based on Genetic Algorithms, Journal of Computer Research and Development, 2001,38(4): 916-922

[7] Rissanen, J. Modeling by shortest data description[J]. Automatica, 1978,14(5):465-471.

[8] Whitley D. The genitor algorithm and selection pressure: why rank-based allocation of reproductive trials is best[C]. // Proceedings of the Third International Conference on Genetic Algorithms. Sanfrancisco: Morgam Kaufm Publish, 1989: 116-121.

[9] Murphy.P.M.& D.W.Aha.UCI repository of machine learning databases. http://www.ics.uci.edu /~mlearn/ MLRepository.html

[10] Cheng, J. PowerConstructor System. http://www.cs. ualberta.ca/~jcheng /bnpc.htm.