Knowledge Discovery of Interesting Classification Rules Based on Adaptive Genetic Algorithm

Yong Zhou[1] Shixiong Xia[1] Dunwei Gong[2] Youwen Li[1]

[1]School of Computer Science & Technology, China University of Mining & Technology, Xuzhou 221008, P. R. China

[2]School of Information & Electrical Engineering, China University of Mining & Technology, Xuzhou 221008, P. R. China

Abstract

Data classification is a very important point in Data Mining, but the existing classification algorithms always only discover the classification rules with high accuracy, and the research about interesting classification rules is few. So this paper proposes an algorithm to find the interesting classification rules based on Genetic Algorithm. Firstly, we design the fitness function with the attributes' information gain, and the settings of weights of the information gain, and the interestingness of the rules, so we combine the objective and subjective measure methods together. Secondly, we use the adaptive genetic algorithm to keep the process from constringency early, and then we can reduce the convergence speed. At last, the results of the experiment given by JBuilder2006 can discover the interesting classification rules, illustrating the effectiveness of this algorithm.

Keywords: Data mining, Genetic algorithm, Classification rules, Interesting rules

1. Introduction

The database technique got a quick development and extensively applied since 70's in 20 centuries, and the data is a valuable fortune which can reflect the development of history, but searching for the useful information from the database is as looking for a needle in a bottom of hay. The absence of efficient method for finding the hidden knowledge induced data rich but knowledge poor, so data miming technique emerged as the times required with the quick development of computer and the requirement of the corporation. Data mining is generally defined as the process of extracting previously unknown, hidden and useful knowledge from a given database [1]. Data classification is an important aspect of data mining. At present, the existing methods of discover classification rules have genetic algorithm, decision tree algorithm, neural network and rough set classification, however, these methods mainly can discover the accurate classification rules, users' inclination can't display by the rules because there's no people join in the classification method. But always the users are interesting at different attributes, so the users hope they can discover the rules which they interested by their own settings. And this paper proposed a method to discover interesting rules base on genetic algorithm, it allow the user join in the evaluation for the rules.

Commonly, the traditional classification methods always can discover the rules with high accuracy, but these high accuracy rules not always are the users need because they also have high universality, and the users may have known them before data classification process, so there rules are no more sense. For example, the rule "if Outlook=Overcast then PlayTennis=Yes" have high accuracy, explaining if as long as its overcast, then can play tennis. It's clear that the rule is easy well known as it don't have high interestingness. On the other hand, the interesting rules usually hide in the database, and these rules with non-predict are always hard to discover, when we get them, they can give the user new views and opinions. The interesting classification rule is defined with interesting, original and new marked character [2]. The research of interesting rules is the key point to study deeply in the domain of science and medicine.

There exist two methods to evaluate the interestingness of the classification rule: the subjective evaluation method and the objective evaluation method. The subjective evaluation method is user-driven and has relationship with the exact problem, but the objective evaluation method is driven by the original data and it's no relationship with the exact problem [2]. The research of Literature [2] found a set of accurate rules firstly on the basis of which further explored the interesting rules, This will ensure the rules has high accuracy, but ignore the subjective evaluation of the rules, and also not easy to discover the rules which have high interestingness but low accuracy; Literature [3] researched the method of the subjective evaluation for the rules; Literature [4]

analyzed the aspects which can affect the quality and the interestingness of rules, including disjunction size of the rules, imbalance of class distributions, attribute interestingness, misclassification costs, and the asymmetric nature of classification rules.

This paper presents an algorithm based on attribute information gain which can combine the subjective evaluation method and objective evaluation method together to discover interesting classification rules. The algorithm allows the users themselves to set the weight of each attribute's information gain, and the weights can reflect they preference, with different weights the algorithm can discover different interesting classification rules.

2. Evaluation of the rules

There exist two methods to evaluate the interestingness of the classification rule at the present literatures: the subjective evaluation method and objective evaluation method. This paper uses the function of attribute information gain to evaluate the rule, and with the setting of the weights of the attribute information gain, it makes the subjective evaluation method and objective evaluation method combine together.

3. The conception and computation of attribute information gain

In all methods of discovering classification rules, the technology of attribute information gain is common used in the decision tree algorithm. The decision tree is basically a greedy algorithm, which use a top-down structure method to make a decision tree. The attribute with the highest information gain which we need has the highest distinction in the given dataset, and it has more classify information, so in the decision tree algorithm, the attribute with the higher of information gain has better effect on classification, and it also has more chance to emerge at the classification rules, namely the attributes must be have higher information gain at the last gained rules [5].

We have to compute the entropy of the dataset before computing each attribute's information gain, as following [5]:

$$Entropy(S) = \sum_{i=1}^{m} -p_i \log_2 p_i$$

Where, m is the number of the class in the dataset S, $p_i \approx \frac{|S_i|}{|S|}$, $|S|$ is the number of the recording in S, and $|S|$ is the number of the recording in class *i*.

Then, we can compute the expectation entropy when uses attribute A to classify the dataset, as following [5]:

$$Entropy(S, A) = \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Where, S_v is a sub-dataset of S which the value of attribute A is *v*.

With the above two formulas, we can compute the attribute A's information gain, as following:

$$InfoGain(S, A) = Entropy(S) - Entropy(S, A)$$

3.1. The objective evaluation for the interesting rules

This paper uses the attribute information gain to evaluate the rule's interestingness objectively, as following [4]:

Definition 1 Rule's Interestingness

$$\text{interestingness} = \frac{1}{\sum_{i=1}^{k} InfoGain(S, A_i)}$$

where, k is the length of the rule, $\sum_{i=1}^{k} InfoGain(S, A_i)$ is the sum of the attributes' information gain, and these attributes must be in this rule, namely, if a rule can be any value at which attribute, then this attribute must be not in the rule, so $\sum_{i=1}^{k} InfoGain(S, A_i)$ don't include this attribute's information gain, and we define the rule's interestingness as its reciprocal.

Definition 1 is based on the idea of the following common sense: the rule with higher information gain will has more chance to be the classification rule as the result of the classification algorithm, and also this rule has more chance to have been known by the user already, namely it has little interestingness. On the contrary, the greedy algorithm has little chance to get the rule with lower attribute information gain, but this rule always is interesting and also is the user's expectation rule. Generally, these rules contain a number of other implied messages that if we use decision tree algorithm for classification, the rule with lower information gain but always has higher interestingness is unlikely to get, so we define the interestingness as its reciprocal, that is to say, the lower of information gain, the higher of the interestingness, then if we use this definition, we can discover the interesting rules.

3.2. The subjective evaluation for the interesting rules

In order to evaluate the rule's interestingness subjectively, we can improve Definition 1.After improved, when it evaluate the rule's interestingness objectively, it will also takes the user's subjective incline. The user may have different interestingness at different attribute when classifying, for example, user 1 may be more careful at attribute A, but user 2 may be more careful at attribute B, so it's necessary to set the weights of the information gain w_i which can show the user's inclination to the attributes, as following:

Definition 2 Rule's Interestingness

$$\text{interestingness}=\frac{1}{\sum_{i=1}^{k} w_i InfoGain(S, A_i)}$$

In Definition 2, we can see that the smaller w_i is, the higher interestingness to the i^{th} attribute is. Fig. 1 illustrates the relationship between user's interestingness and the weight w_i . Compared Definition 1 to Definition 2, it's easy to translate the subjective evaluation method to the objective evaluation method by setting all weights w_i to 1.

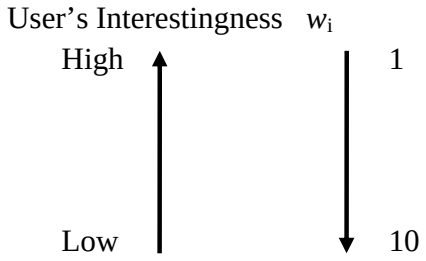

Fig. 1: Relationship between user's interestingness and the weight w_i

4. Knowledge discovery of interesting classification rules based on adaptive genetic algorithm

4.1. The encoding method of classification rules

We must get a method to encode the rule when using the genetic algorithm to discover the classification rules. Generally, the form of the rule with only 1 class attribute is $if\ (v_1=I_1)\wedge(v_2=I_2)\wedge\ldots\wedge(v_n=I_n)$ then c=J, and the part of "if" is the rule's precondition, the "then" part is the result of the rule. This paper uses the Michigan encoding method, namely one chromosome represents one rule, and we don't encode the "class" part, that is to say, all the individuals in initial population are from the same class, and we can get one rule for one class by the algorithm run one time. And if the dataset has n classes, we must run the genetic algorithm n times; certainly we can use the multiprocessor to run at the same time. If the number of the characteristic attributes is n, then the encoding of the rule is as Fig 2.

In Fig. 2, "Value" is the gene's value, it can be one of the. In order to make the length of the chromosome is variable, this paper add a value "Any" to each attribute's value set. When gene's Value="Any", it indicates the corresponding attribute doesn't care about the this value, so we can get rid of this attribute in the rule, then the rule 's length will be minus 1; Weight is the percentage of the recordings whose this attribution is as this rule's account for the whole dataset, we can ignore the attribute which appears a few times by using the Weight part, and the common threshold value is 0.05-0.2; Operator has relationship with the characteristic attribute, it may be "＝"or "≠ " to the disperse characteristic attribute and "≤"or "〉 " to the sequence characteristic attribute; Info Gain is the corresponding attribute's information gain.

GENE1(A1)				·····	GENE(An)			
Weight	Operator	Value	Info Gain	·····	Weight	Operator	Value	Info Gain

Fig. 2: Encoding of the rule

4.2. Fitness function design

The recordings in the dataset can be divided into 4 classes by the rule with the precondition is A and the conclusion is C, as Table 1 following:

In order to get the fitness of the interesting rules, this paper proposed a definition as following:

Definition 3 Rule's Fitness

$$fitness=\frac{pp}{pp+pn}\text{interestingness}$$

Where, $pp+pn$ is the number of the recordings which matching the rule's precondition, pp is the number of the recordings which can completeness

matching the rule, so $\frac{pp}{pp+pn}$ is the rule's accuracy, the larger $\frac{pp}{pp+pn}$ is, the more creditable the rule is. If $\frac{pp}{pp+pn}=1$, namely, pn=0 , that is to say the rule can judge all correctly the recordings which can matching the rule's precondition, but this paper focuses on discovering the interesting rules, so we must import the rule's interestingness besides the accuracy. In order to find the rule with high interestingness and accuracy, we can define the fitness as the product of interestingness and accuracy, then in the evolution process, only the rule with high interestingness and accuracy can survival, so at the last of the algorithm we can the interesting rules.

Rules	Number
If A then C	*pp*
If A then not C	*pn*
If not A then C	*np*
If not A then not C	*nn*

Table 1: The rules and number of matching recordings

4.3. Genetic algorithm's operation

This paper uses the adaptive genetic algorithm to prevent the evolutional process to get premature and reduce the convergence speed. The great advantage of adaptive genetic algorithm is that its crossover rate PC and mutation rate PM are not immutable, but changing with the evolution process [6].

Adaptive genetic algorithm is based on the two following ideas[7]: firstly, when the largest fitness $f_{\max}$ in the group is near to the average fitness f_{avg} of the group, the algorithm should increase PC and PM because the algorithm is trend to be convergence; On the contrary, the algorithm should decrease because the individual in group is diversiform, that is to say PC and PM have inversely proportional to $f_{\max}-f_{avg}$; Secondly, in order to prevent destroying the better gene structure, the individual with higher fitness should have smaller PC and PM, and on the contrary the individual with lower fitness should have larger PC and PM, that is to say PC and PM have directly proportional to $f_{\max}-f$, as the following formulas:

$$P_c=\begin{cases}k_1(f_{\max}-f_{\text{cmax}})/(f_{\max}-f_{avg}), & \text{f}_{\text{cmax}}>\text{f}_{\text{avg}}\\ k_2, & \text{f}_{\text{cmax}}\le \text{f}_{\text{avg}}\end{cases} \quad (1)$$

$$P_m=\begin{cases}k_3(f_{\max}-f_{\text{m}})/(f_{\max}-f_{avg}), & \text{f}_{\text{m}}>\text{f}_{\text{avg}}\\ k_4, & \text{f}_{\text{m}}\le \text{f}_{\text{avg}}\end{cases} \quad (2)$$

In formula (1), $f_{c\max}$ is the higher fitness in the two crossover individuals' fitness, and in formula (2), f_m is the fitness of the individual which will mutate.

From formula (1):

If $f_{c\max}>f_{avg}$, then

$P_c=k_1(f_{\max}-f_{\text{cmax}})/(f_{\max}-f_{avg})<k_1$

If $f_{c\max}\le f_{avg}$, then $P_c=k_2$

The individual with $f_{c\max}>f_{avg}$ has better gene structure, and we don't hope this structure be destroyed, so the crossover rate will be decreased, but on the contrary to the individual with $f_{c\max}\le f_{avg}$, the crossover rate will be increased, so we can get a relationship: $k_1<k_2, k_1,k_2\in(0,1)$

As the same reason, from formula (2), we can get $k_3<k_4$, $k_3,k_4\in(0,1)$

These two formulas explained that the group has more chance to get the local excellent point and get premature if the $f_{\max}-f_{avg}$ is smaller, so the algorithm can increase PC and PM to add the ability for producing new individual. On the contrary, the algorithm will decrease PC and PM to add the convergence ability when $f_{\max}-f_{avg}$ is larger and the group is trend to radiate. Secondly, in order to protect the good gene structure in the same generation, PC and PM must have directly proportional to $f_{\max}-f$.

4.4. Step of the algorithm

The steps of the algorithm for discovering interesting rules are as following:

Step 1 Confirming the characteristic attributes and the class attributes which the classification process need, and get 2/3 recordings of the dataset randomly by the holding method, and these recordings make up of the training dataset T, and the other 1/3 is the testing dataset;

Step 2 Pre-processing the dataset, including data clearing, translating the sequence attribute to the dispersed value, computing each attribute's information gain, and encoding the original group P(t), and setting the counter t =0, the maximum evolutional generation is MAXGEN;

Step 3 Select operator. Firstly, get the fitness of each individual in group P(t) according definition 3, then

carry through the select operator by using roulette select and select kept strategies.

Step 4 Crossover operations. Make the other individuals match together as a couple randomly except the elitist individual from Step 3, and compute each couple's fitness function according definition 3 and the crossover rate P_c according formula (1), then this couple can carry through single point crossover operator.

Step 5 Mutation operations. Firstly, get the fitness of each individual except the elitist individual from Step 3 according definition 3 and the mutation rate P_m according formula (2), then this individual can carry through mutation operator, and update the counter t=t+1;

Step 6 Judgment of the stop condition. If t<MAXGEN, then go to step 3; if t=MAXGEN, then output the individual with highest fitness

5. Experiment and analysis

5.1. The dataset of experiment

In this experiment, we use the Play Tennis dataset [8], as Table2.

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
D1	Rain	Cool	High	Weak	Yes
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Sunny	Mild	Normal	Weak	Yes
D15	Rain	Mild	Normal	Strong	No
D16	Overcast	Hot	High	Strong	Yes
D17	Sunny	Cool	High	Weak	No
D18	Overcast	Cool	Normal	Weak	Yes
D19	Sunny	Hot	Normal	Strong	Yes
D20	Rain	Hot	High	Strong	No
D21	Overcast	Mild	Normal	Weak	Yes
D22	Rain	Hot	High	Weak	Yes
D23	Sunny	Cool	High	Strong	No
D24	Rain	Cool	High	Strong	No
D25	Sunny	Hot	High	Weak	No
D26	Overcast	Mild	Normal	Strong	Yes
D27	Overcast	Cool	High	Strong	Yes
D28	Rain	Mild	High	Strong	No
D29	Sunny	Cool	Normal	Strong	Yes
D30	Overcast	Hot	High	Weak	Yes
D31	Sunny	Hot	High	Strong	No
D32	Rain	Hot	Normal	Strong	No
D33	Overcast	Mild	High	Strong	Yes

Table 2: Play Tennis dataset

From this table, we can find that it includes 33 recordings which divided into 2 classes, and each recording has 4 characteristic attributes and 1 class attribute.

5.2. Parameters setting

The parameters in the experiment are as Table 3.

Parameter	Value
Population Size	11
Chromosome length	4
Maximum Generation	40
Crossover Rate (P_{C0})k2	0.8
k1	0.7
Mutation Rate (P_{M0})k4	0.1
k3	0.09
Threshold of Weight	0.1

Table 3: Parameters setting.

In addition, as an example, this experiment discovered the interesting rules when the user's interesting attribute was "Temperature", and the user set the weights of the information gain according Fig. 1 as following:

$$w1 = 10, w2 = 1, w3 = 10, w4 = 10$$

And the settings can illustrate that the interesting attribute is the second attribute "Temperature" by the weight w2=1.

5.3. Results and analysis

This paper produces the dataset by using holdout method, namely the 2/3 of the recordings in original dataset is chosen randomly for making up of the training dataset, and the other 1/3 will be making up of the testing dataset. And we run the algorithm five times for each class and choose the best rule from these five times as the best rule for the class. Table 4 shows the

rules when the use's interesting attribute is "Temperature" with the settings: $w1=10, w2=1, w3=10, w4=10$

Class	Interesting Rule	The Fitness on Training Set / Testing Set	The Accuracy on Training Set / Testing Set
Yes	if Outlook=Any and Temperature=Mild and Humidity=Any and Wind=Any	34.66 /46.21	0.6 /0.8
No	if Outlook=Any and Temperature=Hot and Humidity=Any and Wind=Any	28.88 /21.66	0.5 /0.375

Table 4: The interesting rule and the fitness and accuracy.

The third column in Table 4 is the fitness of corresponding rule on training set and testing set respectively. Generally, we hope the fitness on training set and testing set is the same that indicates the recordings in original dataset were distributed to training set and testing set averagely, and the gas between the fitness on training set and testing set is caused by the uneven distributing.

The fourth column in Table 4 is the accuracy of corresponding rule on training set and testing set respectively. Generally, we hope the accuracy is close to 1 when discovering the accurate rules, but it's always hard to reach it when discovering the interesting rules and we can regard the accuracy as a ratio which can represent it is appropriate or inappropriate to play tennis. For example the rule of "Yes" class in Table 4, can be explained as the ratio appropriate to play tennis is (0.6+0.8)/2=0.7 when the value of "Temperature" is "Mild".

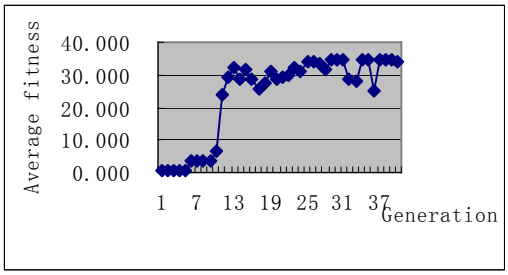

Fig. 3: Relationship of "Yes" Class between Generation and Average Fitness.

Fig.3 shows the relationship between the algorithm's generation and the corresponding average fitness of the "Yes" rule in Table 4. It shows clearly that the fitness is trend to convergence with the increasing of generation and there's an obvious gap between the generation 10 and 12 which indicates the interesting rule emerged, because with the appearance of the interesting rule, the interestingness of the rule will increase rapidly and also the fitness increased.

As the same with Fig.3, Fig.4 shows the relationship between the algorithm's generation and the corresponding average fitness of the "No" rule in Table 4. It shows clearly that the fitness is trend to convergence with the increasing of generation and there's an obvious gap between the generation 21 and 23 which indicates the interesting rule emerged, because with the appearance of the interesting rule, the interestingness of the rule will increase rapidly and also the fitness increased.

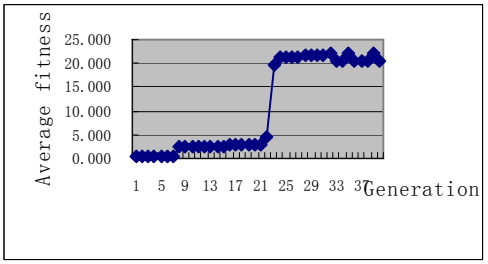

Fig. 4: Relationship of "No" Class between Generation and Average Fitness.

Compared with the original recordings in Table 2, the interesting rules have some use. For example, the user can estimate the ratio of fitting for playing tennis when he only knows the temperature of some day, so the interesting rules can guide the person's actions., By setting the weights of information gain we can find the rules when the user interests the other attributes, but This paper only discovers the interesting rules when the user's interesting attribute is "Temperature" as an example.

6. Conclusions

It's a very worthwhile research topic based on genetic algorithm for discovering interesting rules in data classification, and this paper discussed the issue deeply with proposing a method for evaluating the interesting rule using the attribute information gain and the weight, the advantage of this algorithm is it combines the subjective evaluation and objective evaluation together, and the disadvantage is that the accuracy of the interesting rules is not near to 1, and it costs too much time because we have to run the algorithm for each class especially for the large database, so we will try to implement the algorithm on multiprocessor at future.

Acknowledgement

This work is partially supported by National Natural Science Foundation of China (Grant No. 50674086), National Research Foundation for the Doctoral Program of Higher Education of China (Grant No. 20060290508) and Youth Scientific Research Foundation of China University of Mining and Technology (Grant No. 2006A047).

References

[1] C. Zhou. Gene *Expression Programming and Rule Induction for Domain Knowledge Discovery and Management* .The University of Illinois At Chicago, 2002.

[2] J. Gopalan, R. Alhajj, K. Barker. Discovering Accurate and Interesting Classification Rules Using Genetic Algorithm. *Procsdings of the 2006 International Conference on Data Mining*, pp. 389-395. June 26-29, 2006.

[3] B. Liu， W. Hsu, S. Chen, Using general impression to analyze discovered classification rules. *KDD*, 18(3):31-36, 1997.

[4] A.A. Fretias, On Rule Interesting Measures. *Knowledge Based System*, 12(5): 309-315, 1999.

[5] X. Chen, J.L. Liu, Constructing evaluating indexes system with decision tree method. *Journal of Computer Applications*, 26(2): 368-369, 2006.

[6] L. Wang, T.Z. Shen, Z. Yang. An Improved Adaptive Genetic Algorithm. *Systems Engineering and Electronics*, 24(5): 75-78, 2002.

[7] J. Chen, X.L. Chen, W. Gao, Resampling for Face Detection by Self-Adaptive Genetic Algorithm. *Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004)*, 4: 822-825, August 23-26, 2004.

[8] X.F. Li, J. Li. *Data Mining and Knowledge Discovery*. Beijing: Higher Education Press, 2003.