ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 30 (2012) 457 – 465

Procedia Engineering

www.elsevier.com/locate/procedia

International Conference on Communication Technology and System Design 2011

Multi Objective Optimization of classification rules using Cultural Algorithms

Sujatha Srinivasan[a], Sivakumar Ramakrishnan[a], a*

[a]*Dept. of Computer Science, AVVM Sri Pushpam College, Poondi, Tanjore-613503, Tamil Nadu, India*

Abstract

Classification rule mining is the most sought out by users since they represent highly comprehensible form of knowledge. The rules are evaluated based on objective and subjective metrics. The user must be able to specify the properties of the rules. The rules discovered must have some of these properties to render them useful. These properties may be conflicting. Hence discovery of rules with specific properties is a multi objective optimization problem. Cultural Algorithm (CA) which derives from social structures, and which incorporates evolutionary systems and agents, and uses five knowledge sources (KS's) for the evolution process better suits the need for solving multi objective optimization problem. In the current study a cultural algorithm for classification rule mining is proposed for multi objective optimization of rules.

Keywords: Multi-objective-optimization; Classification; Cultural Algorithm; Data mining.

1. Introduction

Classification rule mining is a class of problems which is the most sought out by decision makers since they produce comprehensible form of knowledge. The rules produced by the rule mining approach are evaluated using various metrics which are called the properties of the rule. The classification rules have to satisfy various properties to be used as a good classifier. The metrics often used are support and confidence. However there are other properties like comprehensibility and interestingness of the rule that make the classifiers more usable. The objectives used for evaluation of rules may sometimes be conflicting. Thus the problem of constructing rules with specific properties should be faced as a multi-objective optimization one [1]. A cultural algorithm (CA) is proposed for multi objective optimization of rules. At least some of the properties of the system should be controlled by the user to make it more

* Sujatha Srinivasan. Tel.: +91-9443884711.
E-mail address; ashok_suja@hotmail.com.

1877-7058 © 2011 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.proeng.2012.01.885

interactive and usable. In the proposed system the user can experiment with the system by specifying certain attributes like the rule metrics (objectives), the rule schema, threshold for the metrics, etc. These user inputs will be used by the agents in the CA for evolving optimized rules.

Cultural algorithm is an evolutionary algorithm which best represents a social system and consists of two levels of evolution: the microevolution in a population space and the macroevolution in a belief space. Through an acceptance function, the experiences of individuals in the population space are used to generate problem solving knowledge that is to be stored in the belief space. The belief space in turn guides the evolution of the population space by means of an influence function. Cultural algorithms have been used for modelling the evolution of complex social systems and for solving numerical optimization problems. The problem and related work on evolutionary data mining for rule induction are discussed in Section 2. Section 3 describes the proposed cultural algorithm which is used for multi objective optimization of rules. Experiments and results are discussed in Section 4. Section 5 concludes with future work.

2. The Problem and Related Work

2.1. The Problem

Given a data source and multiple objectives for optimization namely rule metrics specified by the user, the problem is presenting the user with an optimal and novel set of rules with the specified properties.

Aims of the research: (i). To take data mining in general and rule induction in particular to the next level by incorporating the strengths of Evolutionary computing and Agent technology into Data mining & Knowledge discovery (DM&KD). (ii) The smooth integration of Agent and DM for interactive DM so as to involve users and thus convert knowledge discovered into actionable knowledge.

2.2. Related work

Rule knowledge extraction with specific properties has received little attention in the past years, and typical rule induction algorithms tend to neglect certain desirable properties, such as the ability to induce novel knowledge [1]. Evolutionary algorithms for rule mining as a multi objective optimization problem have been proposed in the literature. Reynolds and Iglesia, [2], [3], [4] use multi-objective algorithms to induce partial classification rules and describe how the use of rule representation and modified dominance relations may increase the diversity of rules presented to the user and how clustering techniques may be used to aid in the presentation of potentially large sets of rules generated. A variety of experiments have been reported by them for partial classification. The algorithm described produce partial classification rules using misclassification cost, rule complexity, support, confidence and coverage as fitness measures for the rules which are allowed to be controlled by the users. Rafael et al. [1] report a research work that combines evolutionary algorithms and ranking composition methods for multi-objective optimization. The candidate solutions are constructed, evaluated and ranked according to their performance in each individual objective, and then rankings are composed into a single rank to solve the multi-objective problem considering all objectives simultaneously. A Multi Objective Evolutionary Algorithm (MOEA) has been proposed by the authors in [5] and [6] for obtaining fuzzy rules for subgroup discovery taking coverage, support and confidence as measures for optimization. The rule induction problem has been considered as a multi-objective combinatorial optimization problem by Khabzaoui et al [7] for finding non frequent and interesting rules using meta-heuristic. A review of evolutionary algorithms for data mining can be found in [8] and for multi objective optimization of classification rules in [9].

Cultural algorithm is a class of evolutionary algorithms which is mostly applied for numerical optimization problems and which has a Knowledge base for representing various primitive knowledge types used by animal species. Lazar and Reynolds, [10] have used CA and rough sets for heuristic knowledge discovery. Sternberg and Reynolds [11] use an evolutionary learning approach based on cultural algorithms to learn about the behavior of a commercial rule-based system for fraud detection. The learned knowledge in the belief space of the cultural algorithm is then used to re-engineer the fraud detection system. Reynolds et al., [12] use decision trees to characterize location decisions made by early inhabitants at Monte Alban, a prehistoric urban center, and have injected these rules into a socially motivated learning system based on cultural algorithms and inferred theories about urban site formation.

3. Extended Cultural Algorithm for multi objective optimization of rules

Cultural algorithms have been used in modeling social systems to solve problems in optimization. But use of cultural algorithm for multi objective optimization of rules is hardly found in the literature. Cultural algorithms use a basic set of knowledge sources, each related to knowledge observed in various animal species. These knowledge sources are then combined to direct the decisions of the individual agents in solving optimization problems. Evolutionary Algorithms are derived from nature and works with a population of individuals in an environment. But they are memory less. Cultural algorithm is a class of evolutionary algorithm that provides a systematic and principled approach for representing knowledge through the five knowledge sources. CA's use knowledge sources to store the various Meta data during the evolution thus incorporating memory into the evolutionary process. Agents evolve using this knowledge in the KS's to produce better individuals. Integration of intelligent agents with Data mining is justified in the article by Cao [13] where the author explores the promising area of agent mining interaction and integration. The current study further integrates evolutionary computing with intelligent agents and data mining to create a complete social system to convert discovered knowledge into social intelligence. Moreover the user can experiment with the system by specifying the various attributes of the system. The following section discusses various parts of the Extended CA (ECA). Thus the proposed system can also be used as a tool kit for experimenting with the process of classification rule mining. Table 1 gives the pseudo code of the cultural algorithm.

Table 1. Pseudo code of cultural algorithm for multi objective optimization of rules

```
Algorithm ECA
Input   : Data Source, rule metrics, parameters
Output: Optimized rules with user specified
properties
Start
        t=0;
        Get Data Set
        Convert data to Chromosome
        Initialize Belief Space, Population
        Space Pop(t)
        Evaluate individuals in Pop(t)
While termination condition is not false do
        t=t+1;
   For each agent do
        Select individual for reproduction using
        social trait
        Reproduce
        Update Pop(t)
   End for
        Evaluate individuals in Pop(t)
        Select Pop(t) from Pop(t-1)
        Accept Pop(t)
        Update belief space
End while
Stop.
```

3.1. The Belief space

The belief space comprises of the five knowledge sources namely the Normative, Situational, Domain, Topographical and the History KS. For the rule optimization problem the five knowledge sources are modified to hold different types of knowledge or Meta data used in solving the problem. Further an additional KS has been added to hold the rules. The agents in the CA have also been given social or cognitive traits which they use in decision making, all of which are described below.

3.1.1. Normative KS

Normative Knowledge Source (NKS) contains the attributes and the possible values that the attribute can take. It gathers this information from the training data set. The normative knowledge source is used to store the maximum and minimum values for numeric attributes. For nominal or discrete attributes, a list of possible values that the attribute can take is stored in the normative KS. The normative KS is used by the agents during mutation.

3.1.2. Situational KS

Situational knowledge source (SKS) consists of the best exemplar found along the evolutionary process. It represents a leader for the other individuals to follow. This way, agents use the leader in-stead of a randomly chosen individual for the recombination. The user can specify a schema which can be used by agents for the search of similar or dissimilar individuals to interest the user.

3.1.3. Domain KS

Domain knowledge source (DKS) contains the vector of rule metrics for each rule. It is updated whenever better rules are accepted into the population at the end of each generation. The domain KS is used by the system to choose best rules for subsequent generations.

3.1.4. Topographical KS

Topographic knowledge source (TKS) is used to store the difference or distance between two rules for the purpose of discovering diverse set of rules to avoid local optima. Hence topographical KS can be used to create novel and interesting rules by using the dissimilarity measure of individuals. This KS is updated at the end of each generation. The topographical knowledge contains a rule pair and their dissimilarity measure.

3.1.5. History KS

History knowledge source (HKS) records in a list, the best individual found at the end of each generation. Evolutionary algorithms are termed as memory less since they do not retain memory of previous generations. However attempts have been made to retain elite individuals of each generation as a separate elite population to render memory to the evolutionary algorithms. Cultural algorithm renders memory to the evolutionary strategy in a systematic way by using the five knowledge sources. History knowledge is used to store best individuals of each generation chosen according to the optimization strategy, thus maintaining memory across generations.

3.1.6. The rule KS

The cultural algorithm is extended by adding another knowledge source namely the Rule KS (RKS) in order to hold the rules. The other KS’s hold a pointer, namely the Rule Id to the rules in the RKS. The rule KS is added to the CA in order to render it to solve the problem of rule mining thus making CA as Extended CA or ECA. The representation of the rule KS is similar to that of the HKS.

3.1.7. Social Agents

The proposed ECA is also extended by adding cognitive traits to the agents. The agents are distinguished by assigning a cognitive trait namely risk taker or imitator or cautious. The agents use this trait in the selection of parents for reproduction using different knowledge sources.

3.2. Influence phase

The influence function decides which knowledge sources influence individuals. In the proposed system this is left to the agents. In the proposed CA the agents use their social trait namely risk taker or imitator or cautious to choose parents for reproduction. Risk takers use knowledge from any of the five knowledge sources at random while cautious agents use only the historical knowledge source. The imitators use the situational knowledge source to create individuals which are similar to the example specified by the user. The normative knowledge source which stores the possible attribute values is used by all the agents during the mutation operation. The topographical knowledge source enables creation of a diverse set of rules. Domain knowledge stores the values of the metrics of the individuals and thus is used for choosing best individuals according to user specified metrics. Thus the five KS's guide the agents in the evolution process.

3.3. Acceptance phase

The acceptance function determines which individuals and their behaviors can impact the belief space. Based on selected parameters such as performance, for example, a percentage of the best performers (e.g., top 10%), can be accepted [12], as in the CA literature. But since the problem is one of classification rule mining, a threshold value for the rule metrics specified by the user is to be used to accept individuals for next generation. The process of agent's selection, reproduction, evaluation forms a generation. At the end of a generation (iteration), the agents return their best individuals along with a vector of rule metric values. The individuals are accepted into the belief space based on the Pareto optimization strategy using the metrics stored in the domain KS as vectors. Dominators which are obtained by the comparison of values of the user specified metrics stored as vectors are chosen for the next generation. The knowledge sources are thus updated at the end of each generation and thus evolve along with the agents. The new values in these KSs then influence the population space. Thus the macro evolution takes place by updating the KSs.

3.4. Evolutionary strategy

Genetic algorithm is by far the most used evolutionary strategy which is also used in the current study. The various attributes of the GA used are discussed below.

3.4.1. Chromosome representation

The chosen data records are converted into chromosomes and represented as a vector of attribute values. The system uses high level encoding where the attribute values are used as they appear in the data source. This reduces the cost of encoding and decoding individuals for creating rules for large data sets. The relational operators are not included in the genotype as found in most algorithms found in the literature. Therefore they are not involved in the reproduction which further minimizes the length of the chromosome and in turn the time taken for encoding and decoding. This representation also avoids use of different types of reproduction operators for different parts of the chromosome. In the current study the class attribute is included in the chromosome. Michigan style rules as disjunction of attribute tests are created only when they are presented to the user.

3.4.2. Population initialization

Population initialization is an important aspect that decides the output of the algorithm. The initial population is created by choosing data sets from the data source at random. This process known as seeding chooses random data from the training data set to be used as seed to create the initial population.

3.4.3. Reproduction operators

The operators used for reproduction are selection, crossover and mutation.

Selection strategy: Unlike algorithms found in the literature, in the proposed CA, Agents use their social traits in choosing the individuals for reproduction as described earlier. In this way, knowledge based selection is used rather than random. This kind of selection strategy aids in creating not only good individuals but also interesting and a diverse set of individuals using the various KS's.

Crossover: One point crossover is used. Initially two individuals are chosen at random from the population. A crossover point is chosen at random and the contents of the chromosome after the crossover point are swapped. Thus crossover takes two parents and produces two children.

Mutation: Mutation operates on individual values of attributes in the chromosome. A mutation point is chosen similar to that of the crossover point which is a random integer. The value of the attribute at that point is replaced by another value depending upon the type of the value. For nominal attributes the value to be replaced is chosen from a list of available values which is also the case of discrete integer values. If the attribute is a continuous valued one, a random value in a specified range of minimum and maximum values is generated and used for reproduction.

3.4.4. Parameters

The parameters that are to be considered and greatly influence the algorithm performance are the crossover rate which is the probability of crossover, and the mutation rate which is the rate of mutation. Also the population size and the number of generations or the termination condition are parameters of importance. Table 2 gives a summary of the parameters used in the experiments.

Table 2. Parameters

Parameters	**Values**
Crossover rate	80%
Mutation rate	20%
Stopping criteria	No. of generations
Initialization process	Seeding
Optimization strategy	Pareto optimality

3.4.5. Optimization strategy

The optimization or multi objective optimization strategy forms the accept phase of the cultural algorithm. This also enables interactive data mining where the user can specify the attributes of the rules to be optimized along with certain threshold values for the various rule metrics. Pareto optimality and ranking composition methods are the frequently used optimization strategies. Pareto optimality has been used in the experiments. Pareto optimality is an optimization strategy that uses comparison of the metrics represented as a vector. An individual "a" is said to be better than another individual "b" if "a" is better than "b" in all the metric values or equal to "b" in all but one metric and better at least in one value. This is enabled by the use of Domain KS which stores the rule metrics as vectors. The entries in the DKS are compared with each other and the best performers in all the metrics are returned as dominators.

4. Experiments and Results

The lenses data set from the UCI Machine Learning Repository [14] has been used to illustrate the problem. The lenses data set has five attributes and 24 instances. Due to its small size it can be used for testing the algorithms in the initial stages. The class attribute takes three values: 1-the patient should be fitted with hard contact lenses, 2-the patient should be fitted with soft contact lenses and 3-patient should not be fitted with contact lenses.

4.1. Experiments

A number of experiments have been carried out for parameter optimization, for testing the performance of the algorithm for multi objective optimization and on various application domains. In the current study the experiments for finding a good value for population size and the number of generations is reported. The population size and number of generations were taken as 100, 200 and 300. The other parameters were kept constant as found in Table 2. The experiments were repeated five times by taking the same train and test data set for each set of parameters. Rules which satisfied 3 out of 5 conditions were selected to be added in the historical KS. However individuals which satisfied all the 4 conditions (LHS of the individuals) were added to the elite population. That is support was taken as the objective to be optimized. Table 3 gives a summary of the values of the two parameters used and the average number of unique rules produced by the algorithm and the number of rules in the elite population. Although the class attribute was included in the training phase, during the test phase classes were assigned to individuals using the following strategy. If all the values in the LHS were satisfied then that class was assigned as the class label. If there was more than one individual covering the LHS then the class with the most number of class counts in the data set was used, else if only three values were equal then that class will be assigned resolving conflicts as above.

Table 3. The number of rules produced by the ECA

Population Size	No. of Generations	Average No. of rules generated	Average No. of Rules in elite population
100	100	65.5	17.6
	200	70.6	19.6
	300	72	21.8
200	100	70.4	16
	200	72	19.2
	300	72	21
300	100	71.4	17.8
	200	71.8	21.2
	300	72	20

4.2. Discussion

Mining rules with desirable properties as specified by the user is considered as a multi objective optimization problem. The proposed Extended Cultural Algorithm uses a genetic algorithm as the evolutionary strategy and Pareto optimality as the optimization criteria. The agents in the ECA use the different knowledge sources to select and produce good rules. It can be seen from Table 3 that the number of rules created by the algorithm ceased to increase after 200 generations and also there was no new rule after the population size was increased to 200 and above. This shows that the value of the population size and number of generations can be varied from 200 to 300. The accuracy of the algorithm on the training set varied from 50% to 78% and on the test set from 50% to 100%. This accuracy value on the training set may be attributed to the fact that the chromosome representation included the class attribute during training which created good individuals but most of which did not satisfy the class attribute condition. This may also be the reason for the more number of rules in the elite population. The performance of the algorithm can be improved by adding a separate procedure for assigning the class attribute to the rule during the training phase. Moreover the results reported here are for a smaller data set with only a few instances and 5 attributes. The algorithm has been observed to perform well on larger data sets but needs testing on high dimensional data sets.

4.3. Performance analysis

The complexity of the algorithm is $O(n^2)$, where n is the number of individuals in the rule KS. The comparisons of individuals to find the similarity value between individuals stored in the TKS and comparisons made to find the dominators using the DKS contributes to the complexity of the algorithm.

4.4. Contributions of the system

The proposed ECA for multi objective optimization of rules contributes in various ways to classification rule mining as well as social computing which are listed below.

i. Parallel rule induction for all the classes simultaneously is enabled by using agents and the KS's.
ii. Interactive Knowledge discovery, i.e. the users can control the various parameters of the system.
iii. The system integrates the two related but seldom studied paradigms of Agent Based Social modeling and Evolutionary data mining using Cultural Algorithm all of which can benefit from each other.
iv. Exploiting the features of evolutionary data mining and social computing to create a complete socially intelligent system.

5. Conclusion and Future work

In the present study an extended cultural algorithm is proposed for multi objective optimization of classification rules, where mining rules with specific properties is taken as a multi objective optimization problem. The extended CA is also improved by incorporating cognitive traits to agents so that the system can also be used as a social system to study the dynamics of an organization or any social system. Further as future work the social knowledge created by the individuals in the ECA is to be converted into actionable social knowledge or collective social intelligence to be applied in an Intrusion detection and prevention system to solve the computer security problem.

References

1. Giusti Rafael, Gustavo E.A., Batista P.A., " Prati Ronaldo Cristiano. Evaluating Ranking Composition Methods for Multi-Objective Optimization of Knowledge Rules", *Proceedings of Eighth International Conference on Hybrid Intelligent Systems*, 2008, p. 537-42.
2. Reynolds A. P. and de la Iglesia, "B. Rule induction using multi-objective metaheuristic: Encouraging rule diversity", *Proceedings of IJCNN*. 2006, p.6375-82.
3. Reynolds A. P. and de la Iglesia B., " Rule Induction for Classification Using Multi-Objective Genetic Programming", In: *Proceedings of 4th Int'l. Conf. on Evolutionary Multi-Criterion Optimization*. LNCS 4403. 2007, p.516-30.
4. Reynolds A. P. and de la Iglesia B., " A Multi-Objective GRASP for Partial Classification", *Soft Computing*, 2009, 13(3): 227-43.
5. Del Jesus M. J., Gonzalez Pedro, and Herrera Francisco, " Multi-objective Genetic Algorithm for Extracting Subgroup Discovery Fuzzy Rules", *Proceedings of the IEEE Symposium on Computational Intelligence in Multi-criteria Decision Making*, 2007, p.50-7.
6. Dehuri S. and Mall R., " Predictive and comprehensible rule discovery using a multi-objective genetic algorithm Knowledge-Based Systems", 2006, 19 :p.413–21
7. Khabzaoui M, Dhaenens C, Talbi EG, " Combining evolutionary algorithms and exact approaches for multi-objective knowledge discovery", *RAIRO Operations Research*, 2008, 42: p. 69–83.
8. Freitas Alex A. , "A Review of Evolutionary Algorithms for Data Mining", *Soft Computing for Knowledge Discovery and Data Mining*, 2007, p.79-111
9. Srinivasan, S. and S. Ramakrishnan, "Evolutionary multi objective optimization for rule mining: a review", *Artificial Intelligence Review*: 2011, 36(3): 205-48, DOI 10.1007/s10462-011-9212-3.

10. Lazar A. and Reynolds R.G., " Heuristic Knowledge Discovery for Archaeological Data Using Cultural Algorithms and Rough Sets", *Heuristics and Optimization for Knowledge Discovery* , Vol., 2 (ed. Ruhul A. Sarker, Hussein A. Abbass, and Charles S. Newton) by Idea Group Publishing, USA, 2002.
11. Sternberg M. and Reynolds R. G., "Using cultural algorithms to support re-engineering of rule-based expert systems in dynamic environments: A case study in fraud detection", *IEEE Trans. Evol. Comput*. 1997, 1(4): p.225–43.
12. Reynolds R. G., Mostafa Ali, and Thaer Jayyousi, " Mining the Social Fabric of Archaic Urban Centers with Cultural Algorithms", *IEEE Computer*, 2008, p.64-72.
13. Cao Longbing, "Introduction to Agent Mining Interaction and Integration, L", *Cao (ed.), Data Mining and Multiagent Integration*, LLC 2009, 2009, p.3-36.
14. Newman D., Hettich S., Blake C. and Merz C., UCI Repository of Machine Learning Databases, Dept. of Information and Computer Science, Univ. of California at Irvine, (1998) http://www.ics.%20uci.edu/~mlearn/MLRepository.html.