ISSN (ONLINE): 2349-7084
GLOBAL IMPACT FACTOR 0.238

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS VOLUME 1, ISSUE 3, SEPTEMBER 2014, PP 127-130

The Use of Heuristics in Decision Tree Learning Optimization

Elma Kolçe (Çela), Neki Frasheri

Abstract— Decision tree algorithms are among the most popular techniques for dealing with classification problems in different areas. Decision tree learning offers tools to discover relationships, patterns and knowledge from data in databases. As the volume of data in databases is growing up very quickly, the process of building decision trees on such databases becomes a quite complex task. The problem with decision trees is to find the right splitting criterion in order to be more efficient and to get the highest accuracy. Different approaches for this problem have been proposed by researchers, using heuristic search algorithms. Heuristic search algorithms can help to find optimal solutions where the search space is simply too large to be explored comprehensively. This paper is an attempt to summarize the proposed approaches for decision tree learning with emphasis on optimization of constructed trees by using heuristic search algorithms. We will focus our study on four of the most popular heuristic search algorithms, such as hill climbing, simulated annealing, tabu search and genetic algorithms.

Index Terms— decision trees, genetic algorithms, heuristics, hill-climbing, simulated annealing, tabu search

1 Introduction

Decision tree induction algorithms represent one of the most popular techniques for dealing with classification problems. Classification, the data mining task of assigning objects to predefined categories, is widely used in the process of intelligent decision making. Many classification techniques have been proposed by researchers in machine learning, statistics and pattern recognition. Decision tree learning offers tools to discover relationships, patterns and knowledge from data in databases. Decision tree induction algorithms (DT) have become widely-used, mainly because the induction of DTs does not require any domain knowledge; DT induction algorithms can handle high-dimensional data; the representation of discovered knowledge in tree form is intuitive and easy to be assimilated by humans; and the learning and classification steps of DT induction are simple and fast [20].

As the volume of data in databases is growing up very quickly, the process of building decision trees on such databases becomes a quite complex task. However, traditional decision tree induction algorithms, such as C4.5 algorithm, implement a greedy approach for node splitting that is inherently susceptible to local optima convergence. Different approaches for this problem have been proposed by researchers, using heuristic search algorithms. Heuristic search algorithms can help to find optimal solutions where the search space is simply too large to be explored comprehensively. Algorithms that either give nearly the right answer or provide a solution not for all instances of the problem are called heuristic algorithms. This group includes a plentiful spectrum of methods based on traditional techniques as well as specific ones. The simplest of search algorithms is exhaustive search that tries all possible solutions from a predetermined set and subsequently picks the best one. Local search is a version of exhaustive search that only focuses on a limited area of the search space. Local search can be organized in different ways. Popular hill-climbing techniques belong to this class. Such algorithms consistently replace the current solution with the best of its neighbors if it is better than the current. Hill-climbing algorithm is effective, but it has a significant drawback called pre-mature convergence. Since it is "greedy", it always finds the nearest local optima of low quality. The goal of modern heuristics is to overcome this disadvantage. Simulated annealing algorithm [21] uses an approach similar to hill-climbing, but occasionally accepts solutions that are worse than the current. The probability of such acceptance is decreasing with time. Tabu search [22] extends the idea to avoid local optima by using memory structures. The problem of simulated annealing is that after "jump" the algorithm can simply repeat its own track. Tabu search prohibits the repetition of moves that have been made recently, saving a lot of time.

2 Hill Climbing and Decision Trees

C4.5 algorithm is one of the most widely used algorithm in the decision trees and so the one of the most popular heuristic function is gain ratio. This heuristic function has a serious disadvantage – towards dealing with irrelevant featured data sources, where it can get trapped to local optima. Hill-climbing can be used as the heuristic function of decision tree, in order to overcome the disadvantage of gain ratio. A.M.Mahmood et al. [14] propose a composite splitting criterion equal to a greedy hill climbing approach and gain ratio. Their experimental results shown that the proposed new heuristic function can scale up accuracy, especially when

- *Author Elma Kolçe (Çela) is currently pursuing PhD degree program in Computer Engineering in Polytechnic University of Tirana, Albania E-mail: elmakolce@yahoo.com*
- *Co-Author Acad.Prof.Dr. Neki Frasheri is currently director of the Center for Research and Development in ICT in Polytechnic University of Tirana, Albania. E-mail: nfrasheri@fti.edu.al*

processing high dimension datasets.

T. De La Rosa et al. [6] also present a novel approach for boosting the scalability of heuristic planners based on automatically learning domain-specific search control knowledge in the form of relational decision trees. Particularly, they define the learning of planning search control as a standard classification process. Then, an off-the shelf relational classifier is used to build domain-specific relational decision trees that capture the preferred action in the different planning contexts of a planning domain. Additionally, they show two methods for guiding the search of a heuristic planner with relational decision trees. The first one consists of using the resulting decision trees as an action policy. The second one consists of ordering the node evaluation of the Enforced Hill Climbing algorithm with the learned decision trees. Their experiments over a variety of domains reveal that in both cases the use of the learned decision trees increase the number of problems solved together with a reduction of the time spent.

3 Simulated Annealing and Decision Trees

The simulated annealing algorithm is a globally optimized algorithm and it can be used to avoid the drawbacks of C4.5. Y.Jiang et al [9] propose a new bank customer credit evaluation method based on decision tree and simulated annealing algorithm. They use C4.5 algorithm to obtain original decision rule sets, and the sets are used as the primary solution of Simulated Annealing Algorithm (SAA) which ensures the high quality data source. With lower target function f(x), SAA makes a better solution decision rules. The process can be modified or abandoned when the decision rules by SAA is too complicated. Their experimental results demonstrate that the proposed method is effective. Later on, the same author [18] also proposes a new credit scoring model based on decision tree and simulated annealing algorithm, which also results effective. J.Dvorak and P. Savicky [13] also use simulated annealing to avoid the drawbacks of C4.5 algorithm. Though it is quite computationally expensive, it allows to adjust the soft thresholds in groups of the nodes simultaneously in a way that better captures interactions between several predictors than the original approach. Their numerical test with data derived from an experiment in particle physics shows that besides the expected better approximation of the training data, also smaller generalization error is achieved.

4 Tabu Search and Decision Trees

Y.Cai et al. [7] use tabu search to investigate the manpower allocation problem with time windows and job-teaming constraints (MAPTWTC), a practical scheduling and routing problem that tries to synchronize workers' schedules to complete all tasks. They first provide an integer programming model for the problem and discuss its properties. Next, they show that tree data structure can be used to represent the MAPTWTC solutions, and its optimal solution can be obtained from one of trees by solving a minimum cost flow model for each worker type. Consequently, they develop a novel tabu search algorithm employing search operators based on the tree data structure. They adopt the standard framework of the tabu search algorithm for the MAPTWTC, considering that there must exist an optimal MAPTWTC solution that can be derived from a tree. So the search space is restricted to a set of feasible trees, from which the optimal solution can be obtained. While K.P.Benneth and J.A.Blue [16] propose an Extreme Point Tabu Search algorithm that constructs globally optimal decision trees for classification problems. Their non-greedy approach minimizes the misclassification error of all the decisions in the tree concurrently. They use Global Tree Optimization to optimize existing decision trees. This capability can be used in data mining for avoiding over fitting, transferring knowledge, incorporating domain knowledge, and maintaining existing decision trees. Their method works by fixing the structure of the decision tree and then representing it as a set of disjunctive linear inequalities. An optimization problem is constructed that minimizes the errors within the disjunctive linear inequalities. To reduce the misclassification error, a nonlinear error function is minimized over a polyhedral region. A new Extreme Point Tabu Search algorithm is used to search the extreme points of the polyhedral region for an optimal solution. Their results are promising in both randomly generated and real-world problems.

5 Genetic Algorithms on Decision Trees

Genetic Algorithms have been widely used to construct short and near-optimal decision trees. In order to utilize genetic algorithms, decision trees must be represented as chromosomes on which genetic operators such as mutation and crossover can be applied. Genetic Algorithms have been used in two ways for finding the near-optimal decision trees. One way is that they can be used to construct decision trees in a hybrid or preprocessing manner (1, 2, 4, 5, 8, and 15). The other way is to apply them directly to decision trees (3, 10, and 12)

S.H.Cha and Ch.Tappert [15] propose utilizing a genetic algorithm to improve the finding of compact, near-optimal binary decision trees. They present a method to encode and decode a decision tree to and from a chromosome where genetic operators such as mutation and crossover can be applied and they also present theoretical properties of decision trees, encoded chromosomes, and fitness functions. Results show that by limiting the tree's height the presented method guarantees finding a better or equal decision tree than the best known algorithms since such trees can be put in the initial population. Y.Modadi et al. [8] also build an accurate classification algorithm with genetic approach that minimizes the number of rules of each classifier and increases classification accuracy. The search space of the algorithm does not check again. If there isn't specification improvement then it is prevented from additional generation. Population size and maximum number of generations is equal to the number of features. The experimental results show that the proposed algorithm achieves the best competitive accuracy compared to other machine learning classification methods. While V.Mohan [1] implements DTs using traditional ID3 algorithm as well as genetic algorithms for learning decision trees. The Traditional Algorithm for learning decision trees is imple-

mented using information gain as well as using gain ratio. Each variant is also modified to combat over fitting using pruning. The Evolutionary Algorithm is implemented with fitness proportionate and rank based as their selection strategy. The algorithm is also implemented to have complete replacement and elitism as replacement strategy. Their results show that the traditional algorithm has performed well in almost all the cases compared to the genetic algorithm, but the training classifier using genetic algorithm takes longer than the traditional algorithm. However GAs are capable of solving a large variety of problems (e.g. noisy data) where traditional decision tree algorithm might fail.

Interesting attempts have been done by R.C.Barros et al. [4]. They show an empirical analysis of a hyper-heuristic evolutionary algorithm that is capable of automatically designing top-down decision-tree induction algorithms. Hyper-heuristics can automatically generate new heuristics suited to a given problem or class of problems by combining, through an evolutionary algorithm, components or building-blocks of human designed heuristics. The proposed hyper-heuristic evolutionary algorithm, HEAD-DT, is a regular generational EA in which individuals are collections of building blocks (heuristics) from decision-tree induction algorithms. It is extensively tested using 20 public UCI data sets and 10 real-world microarray gene expression data sets. The algorithms automatically designed by HEAD-DT are compared with traditional decision-tree induction algorithms, such as C4.5 and CART. Experimental results show that HEAD-DT is capable of generating algorithms which are significantly more accurate than C4.5 and CART.

The other way of using genetic algorithms over decision trees is after the tree is built. Genetic Algorithm has been applied on decision trees, which are already built with C4.5 algorithm, to yield trees of better quality. D.Sh. Liu et al. [3], use a genetic algorithm to optimize the results of decision trees, in order to classify mobile users. The idea of the proposed algorithm, is to use the decision tree algorithm to generate the mobile user classification rules, and then according to the attribute of the rule, such as accuracy, support, simplicity, and gain ratio, they construct the fitness function of genetic algorithm. The larger the value of the fitness is, the more optimal the rule will be. They use the crossover operation and mutation operation of genetic to adjust the fitness function, so the fitness value will reach to the maximum value, and the rule will be optimal. C.J.Hinde et al. [10] extended the capability of decision tree induction systems where the independent variables are continuous. The incremental decision process results inadequate in explaining the structure of several sets of data without enhancement. They add polynomials of the base inputs to the inputs. The polynomials used to extend the inputs are evolved using the quality of the decision trees resulting from the extended inputs as a fitness function that serves to discover structure in the continuous domain resulting in significantly better decision trees.

Fu et al. [12] also use C4.5 Algorithm to construct decision trees, and then they apply genetic algorithm to yield trees of better quality. They conduct a computational study of this approach using a real-life marketing data set, finding that their approach produces uniformly high-quality decision trees and also show that their approach can be used effectively on very large data sets.

6 COMBINATIONS OF HEURISTICS FOR DECISION TREES

Genetic Algorithms have also been combined with other algorithms to build models of decision support systems. W.S.Alsharafat [2] combines Genetic algorithm with Fuzzy Logic to build a generic, multi-criteria model for a decision support system for students' admissions in university. Fuzzy logic method is used for query model representation and Steady State Genetic Algorithms is used for learning the proposed model. This allows to build user-specific and customized models based on information acquired and by interacting with users. They use weight vectors that have linear structures, which can be represented by a binary string in which weight values are converted to binary numbers. These binary strings corresponds to the individual's genes in the GA learning process and they use a specific fitness function for each application, which is computed based on a training data set composed of vectors of fuzzy values. M.Khanbabaei and M.Alborzi [5] also propose a new hybrid mining approach in the design of an effective and appropriate credit scoring model for bank customers. The new proposed hybrid classification model is established based on a combination of clustering, feature selection, decision trees (C4.5), and genetic algorithm techniques. They use clustering and feature selection techniques to preprocess the input samples to construct the decision trees in the credit scoring model. The proposed hybrid model chooses and combines the best decision trees based on the optimality criteria. It constructs the final decision tree for credit scoring of customers. Their results show that the generated decision trees are smaller in size and number of leaves, but have greater classification accuracy compared with other decision tree models.

Simulated Annealing has been used in combination with Tabu Search and Genetic Algorithms to build decision trees. N.Mishra et al. [17] propose a hybrid algorithm named tabu-simulated annealing to solve complex problems of the theory of constraints (TOC), an effective management philosophy for solving decision making problems with the aim of profit maximization by considering the bottleneck in traditional as well as modern manufacturing plants. One of the key components of TOC application is to enumerate quantity of the various products to be manufactured keeping in view the system constraints. Their algorithm exploits the beauty of tabu search and simulated annealing (SA) to ensure the convergence at faster rate. It is found that the performance of hybrid tabu-SA algorithm on a well-known data set of product mix optimization problem is superior as compared to tabu search, SA, TOC heuristic, Revised-TOC (R-TOC) heuristic, and Integer Linear Programming (ILP) based approaches. While R. Ahmed and C.M.Rahman [11] combine the population concept of Evolutionary Algorithms and cooling concept of Simulated Annealing to get control over both speedup and the amount of search space exploration. They propose a new algorithm that utilizes

population-oriented multi-objective simulated annealing (POMOSA) technique for inducing orthogonal decision trees with high predictive accuracy. The new system explores the search space structurally giving priority to the regions that have higher probability of containing better decision trees. The evaluation function they use, is based on two optimization variables, size and pessimistic error and the generated decision trees do not need any pruning. Finally, committee voting over the generated population of better decision trees is used to further increase the predictive accuracy of the system as a whole. Their novel hybrid technique excels by striking a balance between the continued exploration of the problem space and the exploitation of the useful components held in the solutions discovered so far.

7 Conclusions and Future work

Decision tree algorithms are among the most popular techniques for dealing with classification problems, but the problem with them is to find the right splitting criterion in order to get optimal solutions where the search space is simply too large to be explored comprehensively. In this study we have shown that heuristic search algorithms can help in this case. We tried to summarize the proposed approaches for decision tree learning with emphasis on optimization of constructed trees by using four of the most popular heuristic search algorithms, such as hill climbing, simulated annealing, tabu search and genetic algorithms. In each case, we get promising results, better than using traditional decision trees, though algorithms using simulated annealing are quite computationally expensive, so it can be a point to be improved in multi-core systems. Even better results can be taken, by combinations of genetic algorithms and other different heuristics to build more accurate and simpler decisions trees. As future work we aim at building new algorithms based on decision trees and combinations of different types of heuristics trying to employ efficient strategies to cover all the search space, applying local search only in actually promising search areas.

References

[1] V. Mohan, "Decision Trees:A comparison of various algorithms for building Decision Trees", 2013

[2] W.S.Alsharafat "Steady State Genetic Algorithm in University Admission Decision", Contemporary Engineering Sciences, Vol. 6, 2013, no. 5, pp.245 - 254

[3] D.Sh. Liu et al. "A Modified Decision Tree Algorithm Based on Genetic Algorithm for Mobile User Classification Problem", The Scientific World Journal Volume 2014, Article ID 468324 2014

[4] R.C.Barros et al. "Automatic Design of Decision-Tree Algorithms with Evolutionary Algorithms" Massachusetts Institute of Technology 2013

[5] M.Khanbabaei, M.Alborzi "The Use Of Genetic Algorithm, Clustering And Feature Selection Techniques In Construction Of Decision Tree Models For Credit Scoring". International Journal of Managing Information Technology (IJMIT) Vol.5, No.4, 2013 DOI : 10.5121/ijmit.2013.5402

[6] T. de la Rosa et al. "Learning Relational Decision Trees for Guiding Heuristic Planning Association for the Advancement of Artificial Intelligence" (www.aaai.org). 2008

[7] Y.Cai et al. "A Tree-Based Tabu Search Algorithm for the Manpower Allocation Problem with TimeWindows and Job-Teaming Constraints", Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence 2013 pp.496-502

[8] Y.Madadi et al. "An Accurate Classification Algorithm with Genetic Algorithm Approach", International Journal of Computer & Information Technologies (IJOCIT) 2013 ISSN: 2345-3877 Vol 1, Issue 3 pp.198-210

[9] "A Bank Customer Credit Evaluation Based on the Decision Tree and the Simulated Annealing Algorithm" 2007

[10] C.J.Hinde et al. "Evolving the input space for decision tree building" 2005

[11] R.Ahmed et al "Induction of Better Decision Trees Using Population Oriented Multi-Objective Simulated Annealing", 7th International Conference on Computer and Information Technology 2004

[12] Fu Et Al. "A Genetic Algorithm-Based Approach For Building Accurate Decision Trees Informs" Journal On Computing 2003 Vol. 15, No.1, ISSN:1526-5528

[13] J.Dvoˇr´Ak , P.Savick´Y "Softening Splits In Decision Trees Using Simulated Annealing" 2006

[14] A.M.Mahmood et al. "A New Decision Tree Induction Using Composite Splitting Criterion" Journal Of Applied Computer Science & Mathematics, No. 9 (4) /2010

[15] S.H. Cha, Ch.Tappert "A Genetic Algorithm For Constructing Compact Binary Decision Trees" Journal Of Pattern Recognition Research 1 (2009) pp.1-13

[16] K.B. Bennett, J.A.Blue "An Extreme Point Tabu Search Method For Data Mining" 2007

[17] N.Mishra et al., "Hybrid Tabu-Simulated annealing based approach to solve multi-constraint product mix decision problem" Expert Systems with Applications 2005 Volume 29, Issue 2, pp.446-454

[18] Yi Jiang "Credit Scoring Model Based on the Decision Tree and the Simulated Annealing Algorithm" Computer Science and Information Engineering, 2009 WRI World Congress Vol.4, pp. 18 – 22, ISBN: 978-0-7695-3507-4

[19] Glover, F. & Laguna, M. 1997. Tabu search. Boston: Kluwer Academic Publishers.

[20] J. Han, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.

[21] M. E. Aydin, T. C. Fogarty. "A Distributed Evolutionary Simulated Annealing Algorithm for Combinatorial Optimization Problems", in Journal of Heuristics 2004, vol. 24, no. 10, pp. 269–292.

[22] R. Battiti. "Reactive search: towards self-tuning heuristics", in Modern heuristic search methods. Wiley&Sons, 1996, pp. 61-83.

[23] Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Company, Inc., Reading, MA