Information Sciences 221 (2013) 403–418

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Dynamic programming approach to optimization of approximate decision rules

Talha Amin [a], Igor Chikalov [a], Mikhail Moshkov [a,*], Beata Zielosko [a,b,*]

[a] Mathematical and Computer Sciences & Engineering Division, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia
[b] Institute of Computer Science, University of Silesia, 39, Będzińska St., 41-200 Sosnowiec, Poland

ARTICLE INFO

Article history:
Received 29 August 2011
Received in revised form 8 August 2012
Accepted 18 September 2012
Available online 28 September 2012

Keywords:
Approximate decision rules
Dynamic programming
Length
Coverage

ABSTRACT

This paper is devoted to the study of an extension of dynamic programming approach which allows sequential optimization of approximate decision rules relative to the length and coverage. We introduce an uncertainty measure $R(T)$ which is the number of unordered pairs of rows with different decisions in the decision table T. For a nonnegative real number β, we consider β-decision rules that localize rows in subtables of T with uncertainty at most β. Our algorithm constructs a directed acyclic graph $\Delta_\beta(T)$ which nodes are subtables of the decision table T given by systems of equations of the kind "attribute = value". This algorithm finishes the partitioning of a subtable when its uncertainty is at most β.

The graph $\Delta_\beta(T)$ allows us to describe the whole set of so-called irredundant β-decision rules. We can describe all irredundant β-decision rules with minimum length, and after that among these rules describe all rules with maximum coverage. We can also change the order of optimization. The consideration of irredundant rules only does not change the results of optimization. This paper contains also results of experiments with decision tables from UCI Machine Learning Repository.

1. Introduction

Decision rules are widely used as parts of algorithms (parallel or nondeterministic), as a way for knowledge representation, and also as parts of classifiers (predictors). Exact decision rules can be overfitted, i.e., depend essentially on the noise or adjusted too much to the existing examples.

Approximate rules are less dependent on the noise. They have usually smaller number of attributes and are better from the point of view of understanding. Classifiers based on approximate decision rules have often better accuracy than the classifiers based on exact decision rules. Therefore approximate decision rules and closely connected with them approximate reducts are studied intensively last years by Nguyen et al. [5,9,10,20–23,25,26,28,31,32,34,35,37].

There are different approaches to the construction of decision rules and reducts: brute-force approach which is applicable to tables with relatively small number of attributes, genetic algorithms [3,35,36], Apriori algorithm [1], simulated annealing [15], Boolean reasoning [22,27,33], separate-and-conquer approach (algorithms based on a sequential covering procedure) [6,11,13,14,16], ant colony optimization [7,17,24], algorithms based on decision tree construction [18,21,29], different kinds of greedy algorithms [20,22]. Each method can have different modifications. For example, as in the case of decision trees, we

* Corresponding authors. Address: Institute of Computer Science, University of Silesia, 39, Będzińska St., 41-200 Sosnowiec, Poland (B. Zielosko).
E-mail addresses: talha.amin@kaust.edu.sa (T. Amin), igor.chikalov@kaust.edu.sa (I. Chikalov), mikhail.moshkov@kaust.edu.sa (M. Moshkov), beata.zielosko@us.edu.pl (B. Zielosko).

0020-0255/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2012.09.018

can use greedy algorithms based on different uncertainty measures (Gini index, entropy, etc.) for construction of decision rules.

We are interested in the construction of short rules which cover many objects. In particular, the choice of short rules is connected with the Minimum Description Length principle [30]. The rule coverage is important to discover major patterns in the data. Unfortunately, the problems of minimization of length and maximization of coverage of decision rules are NP-hard. Using results of Feige [12] it is possible to prove that under reasonable assumptions on the class NP there are no approximate algorithms with high accuracy and polynomial complexity for minimization of decision rule length.

The most part of approaches mentioned above (with the exception of brute-force, Apriori, and Boolean reasoning) cannot guarantee the construction of shortest rules or rules with maximum coverage.

To work with approximate decision rules, we introduce an uncertainty measure $R(T)$ that is the number of unordered pairs of rows with different decisions in the decision table T. Then we consider β-decision rules that localize rows in subtables of T with uncertainty at most β.

To construct optimal β-decision rules we use approach based on an extension of dynamic programming. We construct a directed acyclic graph $\Delta_\beta(T)$ which nodes are subtables of the decision table T given by systems of equations of the kind "attribute = value". We finish the partitioning of a subtable when its uncertainty is at most β. The parameter β helps us to control computational complexity and makes the algorithm applicable to solving more complex problems.

The constructed graph allows us to describe the whole set of so-called irredundant β-decision rules. Then we can find (in fact, describe) all irredundant β-decision rules with minimum length, and after that among these rules find all rules with maximum coverage. We can change the order of optimization: find all irredundant β-decision rules with maximum coverage, and after that find among such rules all irredundant β-decision rules with minimum length.

We prove that by removal of some conditions from the left-hand side of each β-decision rule that is not irredundant and by changing the decision on the right-hand side of this rule we can obtain an irredundant β-decision rule which length is at most the length of initial rule and the coverage is at least the coverage of initial rule. It means that we work not only with optimal rules among irredundant β-decision rules but also with optimal ones among all β-decision rules.

Similar approach to decision tree optimization was considered in [2,8,19]. First results for decision rules based on dynamic programming methods were obtained in [38]. The aim of this study was to find one decision rule with minimum length for each row.

In this paper, we consider algorithms for optimization of β-decision rules relative to the length and coverage, and results of experiments with some decision tables from UCI Machine Learning Repository [4] based on Dagger software system created in KAUST.

This paper consists of nine sections. Section 2 contains main notions. Section 3 is devoted to the consideration of irredundant β-decision rules. In Section 4, we study directed acyclic graph $\Delta_\beta(T)$ which allows to describe the whole set of irredundant β-decision rules. In Section 5, we consider a procedure of optimization of this graph (in fact, corresponding β-decision rules) relative to the length, and in Section 6 – relative to the coverage. In Section 7, we discuss possibilities of sequential optimization of rules relative to the length and coverage. Section 8 contains results of experiments with decision tables from UCI Machine Learning Repository, and Section 9 – conclusions.

2. Main notions

In this section, we consider definitions of notions corresponding to decision tables and decision rules.

A *decision table* T is a rectangular table with n columns labeled with conditional attributes $f_1, \ldots, f_n$. Rows of this table are filled by nonnegative integers which are interpreted as values of conditional attributes. Rows of T are pairwise different and each row is labeled with a nonnegative integer (decision) which is interpreted as a value of the decision attribute.

We denote by $N(T)$ the number of rows in the table T. By $R(T)$ we denote the number of unordered pairs of rows with different decisions. We will interpret this value as *uncertainty* of the table T.

The table T is called *degenerate* if T is empty or all rows of T are labeled with the same decision. It is clear that in this case $R(T) = 0$.

A minimum decision value which is attached to the maximum number of rows in T will be called the *most common decision for* T.

Let $f_{i_1}, \ldots, f_{i_m} \in \{f_1, \ldots, f_n\}$ and $a_1, \ldots, a_m$ be nonnegative integers. We denote by $T(f_{i_1}, a_1) \ldots (f_{i_m}, a_m)$ the subtable of the table T which contains only rows of T that have numbers $a_1, \ldots, a_m$ at the intersection with columns $f_{i_1}, \ldots, f_{i_m}$. Such subtables (including the table T) are called *separable subtables* of T.

We denote by $E(T)$ the set of attributes from $\{f_1, \ldots, f_n\}$ which are not constant on T. For any $f_i \in E(T)$, we denote by $E(T, f_i)$ the set of values of the attribute f_i in T.

The expression

$$f_{i_1} = a_1 \wedge \ldots \wedge f_{i_m} = a_m \rightarrow d \tag{1}$$

is called a *decision rule over* T if $f_{i_1}, \ldots, f_{i_m} \in \{f_1, \ldots, f_n\}$, and $a_1, \ldots, a_m, d$ are nonnegative integers. It is possible that $m = 0$. In this case (1) is equal to

$$\rightarrow d.$$