Knowledge-Based Systems 56 (2014) 68–78

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Knowledge reduction for decision tables with attribute value taxonomies

Mingquan Ye [a,c,*], Xindong Wu [a,b], Xuegang Hu [a], Donghui Hu [a]

[a] Department of Computer Science, Hefei University of Technology, Hefei 230009, PR China
[b] Department of Computer Science, University of Vermont, Burlington, VT 05405, USA
[c] Department of Computer Science, Wannan Medical College, Wuhu 241002, PR China

ARTICLE INFO

Article history:
Received 5 December 2012
Received in revised form 17 October 2013
Accepted 29 October 2013
Available online 11 November 2013

Keywords:
Knowledge reduction
Attribute value taxonomy
Attribute generalization
Classification
Rough set theory

ABSTRACT

Attribute reduction and attribute generalization are two basic methods for simple representations of knowledge. Attribute reduction can only reduce the number of attributes and is thus unsuitable for attributes with hierarchical domains. Attribute generalization can transform raw attribute domains into a coarser granularity by exploiting attribute value taxonomies (AVTs). As the control of how high an attribute should be generalized is typically quite subjective, it can easily result in over-generalization or under-generalization. This paper investigates knowledge reduction for decision tables with AVTs, which can objectively control the generalization process, and construct a reduced data set with fewer attributes and smaller attribute domains. Specifically, we make use of Shannon's conditional entropy for measuring classification capability for generalization and propose a novel concept for knowledge reduction, designated attribute-generalization reduct, which can objectively generalize attributes to maximize high levels while keep the same classification capability as the raw data. We analyze major relationships between attribute reduct and attribute-generalization reduct and prove that finding a minimal attribute-generalization reduct is an NP-hard problem and develop a heuristic algorithm for attribute-generalization reduction, namely, AGR-SCE. Empirical studies demonstrate that our algorithm accomplishes better classification performance and assists in computing smaller rule sets with better generalized knowledge compared with the attribute reduction method.

1. Introduction

Real-world applications in data mining and machine learning require that their tools can reduce dimensionality of large databases and build classifiers with improved classification performance [1,25]. In practice, data are often represented in the form of decision tables with not only a huge number of attributes but also large cardinalities of attribute domains. Utilizing most classification methods, such as rule-based classifiers, on such data is rather difficult and often infeasible [19,38]. It is therefore desirable to develop data preprocessing techniques to reduce both the number of attributes and the cardinalities of attribute domains and generate more accurate and compact classifiers from reduced data [19,36].

One of the key issues of preprocessing techniques is knowledge reduction. Different methods have been proposed for effective and efficient reduction of knowledge [1,4,5,24]. Of all the paradigms, rough set theory (RST), proposed by Pawlak [21], is a relatively new soft computing tool for analyzing various types of data and makes significant contributions to this field [7,8,22]. Attribute reduction in RST offers a systematic theoretic framework for finding particular subsets of condition attributes that provide the same descriptive or classification ability as the entire set of attributes. However, this reduction method can only reduce the number of attributes and is thus unsuitable for attribute domain reduction.

One method in data mining is to provide domain reduction of attributes in a database by generalization [3,9]. This approach is known as attribute-oriented induction (AOI). Attribute generalization is achieved by using an attribute value taxonomy (AVT, also known as concept hierarchy). Such an AVT reflects necessary background knowledge, which controls the generalization process, and can range from the single, most generalized root concept to the most specific concepts corresponding to the specific values of attributes in the database [3,9,10,36].

In the AOI method, the generalization process is performed by either attribute removal or attribute generalization and guided by two thresholds: the attribute threshold and the relation threshold [3,10,36]. The attribute threshold specifies the maximum number of distinct values of any attribute that may exist after generalization, and the relation threshold provides an upper bound to the number of generalized objects that remain after the generalization process. However, the thresholds are often provided by domain experts or knowledge engineers, and the control of how high an attribute should be generalized is typically quite subjective,

* Corresponding author at: Department of Computer Science, Hefei University of Technology, Hefei 230009, PR China. Tel.: +86 13956155965.
E-mail addresses: ymq@wnmc.edu.cn (M. Ye), xwu@cs.uvm.edu (X. Wu), jsjxhuxg@hfut.edu.cn (X. Hu), hudh@hfut.edu.cn (D. Hu).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.10.022

which may lead to over-generalization or under-generalization [3,9,10,26]. Therefore, it is interesting to objectively calculate the corresponding abstraction level that each attribute value should be generalized to.

In an AVT, the root is the most abstract value "ANY" of an attribute. Leaf nodes are attribute values appearing in the given raw table, and internal nodes represent generalized attribute values of their child nodes. A generalization replaces some values with a parent value in the AVT. With respect to classification tasks, there are two generalization schemes for domain consistency: full-domain generalization (FDG) [2,6,11,12] and full-subtree generalization (FSG) [34–37]. In FDG, all values in an attribute are generalized to the same level of the AVT. In the FSG, at a non-leaf node, either all child values or none are generalized, and a generalized attribute has values that form a "cut" through its AVT. AVT may be defined for either a discrete or continuous valued attribute. A leaf concept for a continuous attribute is expressed as a range of values.

Example 1. Consider a flat decision table with condition attribute set $C = \{Job, Age, Sex\}$ and decision attribute set $D = \{Salary\}$ in Fig. 1a. The table has 12 records in total. Each row means one or more records with the *Salary* column including the class frequency of the records represented. "L" means their Salary is Low, and "H" means their Salary is High. Fig. 1b shows the AVTs of C, Fig. 1c and d show the FDG and FSG for the AVT of attribute *Job*, respectively. According to FDG, if "Engineer" and "Lawyer" are generalized to "Professional", then it also requires generalizing "Dancer" and "Writer" to "Artist". While according to FSG, if "Engineer" is generalized to "Professional", this scheme also requires the other child node, "Lawyer" to be generalized to "Professional", but "Dancer" and "Writer", which are child nodes of Artist, can remain ungeneralized.

Recently, RST has been used to mine generalized decision rules from decision tables and hierarchical attributes [6,11,12]. However, they still lack the adaptability in solving attribute generalization. In particular, their work adopted the FDG on the AVT and caused the largest distortion of the data for the same granularity level requirement on all paths of a taxonomy tree. AVT with the FSG is more general and provides a way to organize data at different levels of granularity, which have been demonstrated to be useful in generating accurate, compact, and comprehensible classifiers [34–37].

According to the above observations, we focus on the issues involved with FSG when we are faced with decision tables and attribute value taxonomies (AVTs) for condition attributes and generalizing condition attribute values to maximize concept levels while retaining the classification ability at primitive level. Wang [29,30] used Shannon's conditional entropy to measure the classification ability of condition attributes with respect to decision attributes and constructed a heuristic algorithm for attribute reduction. This reduction method keeps the conditional entropy of target decision unchanged. Therefore, we use it to measure classification capability of different generalization levels of condition attributes. Paralleling attribute reduct for a flat data table, we present a novel concept for knowledge reduction in decision tables and attributes associated with AVTs, namely attribute-generalization reduct.

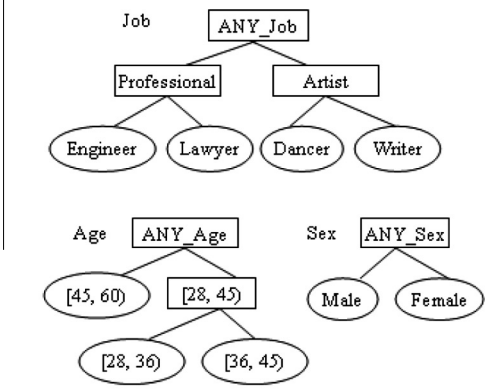
Attribute-generalization reduction can objectively calculate the corresponding abstraction level that each attribute value should be generalized to and induce the most abstract generalized decision table with the same classification ability on the raw decision table, and no other generalized decision table exists that is more abstract than it.

We discuss the relationship between attribute-generalization reduct and attribute reduct. Generally speaking, the reduced data generated by attribute reduction methods are composed of primitive values, while the reduced data generated by attribute-generalization reduction methods are composed of primitive values, as well as generalized values from the AVTs. Therefore, we can think of attribute reduct as a special case of attribute-generalization reduct where for each attribute, a two-level hierarchy with the root node of "ANY" and leaf nodes corresponding to all attribute values in the raw decision table.

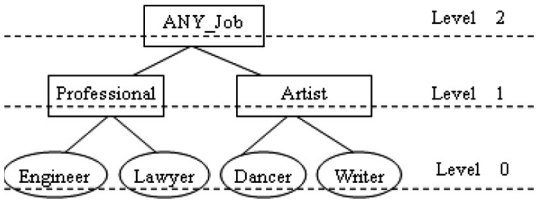
We prove that the problem of attribute-generalization reduct generation is NP-hard and develop a heuristic algorithm for attribute-generalization reduction. To evaluate our algorithm, we conducted experiments with datasets from the UCI machine

U	Job	Age	Sex	Salary
x_1	Engineer	[28, 36)	Male	0H,1L
x_2	Engineer	[28, 36)	Female	0H,1L
x_3	Lawyer	[28, 36)	Female	1H,0L
x_4	Lawyer	[28, 36)	Male	1H,0L
x_5	Lawyer	[36, 45)	Male	1H,0L
x_6	Engineer	[36, 45)	Male	0H,1L
x_7	Dancer	[28, 36)	Female	1H,0L
x_8	Dancer	[45, 60)	Female	0H,1L
x_9	Dancer	[36, 45)	Male	1H,0L
x_{10}	Writer	[45, 60)	Male	0H,1L
x_{11}	Writer	[36, 45)	Male	1H,0L
x_{12}	Writer	[45, 60)	Female	0H,1L
Total				6H,6L

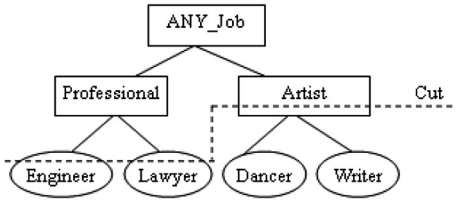
(a) A raw decision table S



(b) Attribute value taxonomies on {*Job*, *Age*, *Sex*}



(c) Full-domain generalization on *Job*



(d) Full-subtree generalization on *Job*

Fig. 1. Different generalization schemes of the decision table and AVTs.