Greedy Algorithm for Modeling Approximate Decision Trees for Distributed Decision Tables

Azimkhon Ostonov[1[0000−0001−5763−9751]], Mikhail Moshkov[1[0000−0003−0085−9483]], and Beata Zielosko[1,2[0000−0003−3788−1094]]

[1] Computer, Electrical and Mathematical Sciences & Engineering Division
King Abdullah University of Science and Technology (KAUST)
Thuwal 23955-6900, Saudi Arabia
{azimkhon.ostonov,mikhail.moshkov}@kaust.edu.sa
[2] Institute of Computer Science, University of Silesia in Katowice
Bedzinska 39, Sosnowiec, 41-200, Poland
beata.zielosko@us.edu.pl

Abstract. This paper addresses the problem of construction an approximate decision tree with minimum depth for distributed data represented as a tuple of decision tables. Unfortunately this is an NP-hard problem, so a greedy algorithm has been proposed which induces approximate, so-called shared decision tree. Theoretical results related to the bounds of the depth of decision trees constructed by the proposed greedy method are presented.

Keywords: Distributed Data · Shared Decision Tree · Greedy Algorithm.

1 Introduction

Decision trees [4, 5] are extensively employed for knowledge representation, classification, and problem solving in fields such as fault diagnosis and combinatorial optimization. There are a significant number of results related to the bounds on the complexity of decision trees [1, 3, 6, 9]. Additionally, a wide range of algorithms and their different variants have been developed during many years.

When data is collected in a single decision table, constructing a decision tree is a relatively simple task. However, constructing a decision tree for a set of decision tables is a much greater challenge. In the paper, unlike the usual approach to working with decision trees, we build a tree not for one, but for several decision tables simultaneously.

This paper continues the series begun in [10] where problems related to multi-agent decision making are considered. There are n agents, each of which collects data and represents it in the form of a decision table. The agents solve somewhat similar problems and use attributes from a common set. The tuple of the considered n decision tables is called a distributed decision table.

In this paper, we consider the case where all decision tables are available. Given a tuple of values of all attributes present in the tables, we need for each

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

table to find a decision or show that there is no matching row. There are two options: either build a decision tree for each table and use them sequentially (we are only considering a single-processor environment) or build one shared decision tree that is valid for all decision tables simultaneously.

We explore the second option and, in addition, study approximate shared decision trees. Since the number of nodes in such trees can grow exponentially with the total number of elements in the decision tables, we do not construct the entire tree but instead simulate its operation with a given set of attribute values. We show that even in this formulation, the problem of describing an approximate shared decision tree with minimum depth is NP-hard. Therefore, we focus on a greedy algorithm for constructing an approximate shared decision tree. This algorithm is a generalization of the greedy algorithm applicable to a single decision table [7]. We obtain precision bounds for the depth of decision trees constructed by the new greedy algorithm.

The considered problem is a form of distributed data mining [2], where a global decision model (shared decision tree) is constructed from multiple distributed decision tables associated with different agents.

The rest of the paper is organized as follows. Section 2 contains the main notions concerning proposed new approach and Sect. 3 – three auxiliary statements. Section 4 is devoted to the main results and Sect. 5 – to the study of distributed decision tables related to the point color recognition. Section 6 contains conclusions and future works.

2 Main Notions

In this section, we introduce the main notions related to distributed decision tables and approximate shared decision trees constructed for them, and we formulate the problem of minimizing the depth of an α-decision tree.

Let $k \geq 2$. A *k-valued decision table* (or simply, a *decision table*) U is a rectangular table whose entries belong to the set $E_k = \{0, \ldots, k-1\}$. The columns of this table are labeled by pairwise distinct attributes (i.e., attribute names) from the set $\{g_1, g_2, \ldots\}$. The rows of U are pairwise distinct, and each row is labeled with a natural number, referred to as a decision. We denote by $At(U)$ the set of attributes assigned to the columns of U.

The table U is called *degenerate* if it is empty (contains no rows) or if all of its rows share the same decision; otherwise, it is called *nondegenerate*. A decision that appears in the greatest number of rows of U is called the *most common decision of U*. If several such decisions exist, we choose the smallest one. If U is empty, its most common decision is defined to be 1.

A *subtable* of U is any table obtained from U by deleting some of its rows.

A *distributed decision table* is a tuple $\mathcal{U} = (U_1, \ldots, U_n)$ of $n \geq 1$ decision tables $U_1, \ldots, U_n$. Denote $At(\mathcal{U}) = \bigcup_{j=1}^{n} At(U_j)$. Let, for the definiteness, $At(\mathcal{U}) = \{g_1, \ldots, g_m\}$. The distributed decision table $\mathcal{U} = (U_1, \ldots, U_n)$ is called *degenerate* if tables $U_1, \ldots, U_n$ are degenerate, and *nondegenerate* otherwise. We correspond to a distributed decision table $\mathcal{U} = (U_1, \ldots, U_n)$ a tuple $(s_1, \ldots, s_n)$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

where, for $j = 1, \ldots, n$, the value s_j is equal to the symbol $\emptyset$ if the table U_j is empty and s_j is the most common decision for U_j otherwise. This tuple will be called the *general decision* for $\mathcal{U}$. A *subtable* of $\mathcal{U}$ is a distributed decision table $\mathcal{U}' = (U_1', \ldots, U_n')$ such that U_j' is a subtable of U_j for $j = 1, \ldots, n$.

We now define the parameter $R(\mathcal{U})$. Let U_j be a table from $\mathcal{U}$. We denote by $R(U_j)$ the number of unordered pairs of rows from U_j labeled with different decisions. Then $R(\mathcal{U}) = R(U_1) + \cdots + R(U_n)$. Evidently, $R(\mathcal{U}) = 0$ if and only if $\mathcal{U}$ is degenerate.

We denote by $\Omega(\mathcal{U})$ the set of finite words over the alphabet $\{(g_i, \sigma) : g_i \in \{g_1, \ldots, g_m\}, \sigma \in E_k\}$ including the empty word λ. Let $\beta \in \Omega(\mathcal{U})$ and U_j be a table from $\mathcal{U}$. We now define a subtable $U_j\beta$ obtained from U_j by the removal of some rows. If $\beta = \lambda$, then $U_j\beta = U_j$. Let $\beta \neq \lambda$, $\beta = (g_{l_1}, \sigma_1) \ldots (g_{l_k}, \sigma_k)$ and r be a row of U_j. Then this row is removed if and only if for some $t \in \{1, \ldots, k\}$, $g_{l_t} \in At(U_j)$ and in the intersection of the row r and the column labeled with the attribute g_{l_t} there is a number different from σ_t. Denote $\mathcal{U}\beta = (U_1\beta, \ldots, U_n\beta)$.

A *decision tree Γ over $\mathcal{U}$* is a finite rooted tree in which each terminal node is labeled with an n-tuple consisting of natural numbers and the symbol $\emptyset$, and each nonterminal (i.e., *working*) node is labeled with an attribute from $At(\mathcal{U}) = \{g_1, \ldots, g_m\}$. Exactly k edges leave every working node, and these edges are labeled with pairwise distinct elements of E_k.

To every terminal node v in the tree Γ we associate a word $\beta(v)$. If v is the root of Γ, then $\beta(v) = \lambda$. If v is not the root, and the path from the root to v passes through $p > 0$ nodes labeled with attributes $g_{i_1}, \ldots, g_{i_p}$, along edges labeled with $\sigma_1, \ldots, \sigma_p$, then we define

$$\beta(v) = (g_{i_1}, \sigma_1) \cdots (g_{i_p}, \sigma_p).$$

Let α be a real number with $0 \leq \alpha < 1$. We say that Γ is a *shared α-decision tree for $\mathcal{U}$* (or simply, an *α-decision tree for $\mathcal{U}$*) if, for every terminal node v of Γ, the inequality

$$R(\mathcal{U}\beta(v)) \leq \alpha\, R(\mathcal{U})$$

is satisfied, and the node v is labeled with the general decision for $\mathcal{U}\beta(v)$.

Let Γ be an α-decision tree for $\mathcal{U}$. For any tuple $\bar{\sigma} \in E_k^m$ representing the values of attributes $g_1, \ldots, g_m$, the tree operates as follows. We start at the root of Γ. If the current node is terminal, then the output of Γ is the general decision assigned to that node. If the current node is a working node labeled with attribute g_i, and the value of g_i in the tuple is $\sigma \in E_k$, then we follow the outgoing edge labeled with σ, and continue this process accordingly.

It is clear that, for each tuple $\bar{\sigma} = (\sigma_1, \ldots, \sigma_m) \in E_k^m$, in the tree Γ there exists a terminal node v such that the word $\beta(v)$ is a word over the alphabet $\{(g_1, \sigma_1), \ldots, (g_m, \sigma_m)\}$ for which $R(\mathcal{U}\beta(v)) \leq \alpha R(\mathcal{U})$. We will say that the *node v accepts the tuple $\bar{\sigma}$*.

Let us fix α such that $0 \leq \alpha < 1$. We now consider an example, which shows that the minimum number of nodes in an α-decision tree for $\mathcal{U}$ can grow exponentially with the total number of elements in the tables $U_1, \ldots, U_n$. The

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

number of elements in the decision table is equal to the product of the number of rows and the number of columns.

Example 1. Let us consider a distributed decision table $\mathcal{U}^* = (U_1^*, \ldots, U_n^*)$ in which, for $j = 1, \ldots, n$, $At(U_j^*) = \{g_j\}$ and the table U_j^* is nondegenerate 2-valued table. The table U_j^* contains one column labeled with g_j and two rows (0) and (1) labeled with different decisions. The number of elements in this decision table is equal to 2. The total number of elements in the decision tables $U_1^*, \ldots, U_n^*$ is equal to $2n$. We have $At(\mathcal{U}^*) = \{g_1, \ldots, g_n\}$.

Let Γ^* be an α-decision tree for $\mathcal{U}^*$. We denote by V the set of terminal nodes of Γ^* each of which accepts at least one tuple from E_2^n. Let $v \in V$ and $\bar{\sigma} = (\sigma_1, \ldots, \sigma_n) \in E_2^n$ be a tuple accepted by the node v. It is clear that the word $\beta(v)$ should contain at least $(1-\alpha)n$ pairwise different letters from the set $\{(g_1, \sigma_1), \ldots, (g_n, \sigma_n)\}$: otherwise, $R(\mathcal{U}^*\beta(v)) > \alpha R(\mathcal{U}^*)$. Therefore, v accepts at most $2^{\lfloor \alpha n \rfloor}$ tuples from E_2^n. Thus, $|V| \geq 2^{n - \lfloor \alpha n \rfloor} \geq 2^{n(1-\alpha)}$. It means that the decision tree Γ^* contains at least $2^{n(1-\alpha)}$ nodes.

This example demonstrates that instead of constructing an entire decision tree, it is more appropriate to model its operation on a given tuple of attribute values from $At(\mathcal{U})$.

We denote by $d(\Gamma)$ the *depth* of a decision tree Γ, defined as the maximum length of any path from the root to a terminal node. We write $d_\alpha(\mathcal{U})$ for the minimum possible depth of an α-decision tree for the distributed table $\mathcal{U}$.

We are interested in minimizing the depth of α-decision trees for distributed decision tables. To this end, we will consider the following *problem of α-decision tree depth minimization*. Given a nondegenerate distributed decision table $\mathcal{U}$ and a given tuple of values of attributes from the set $At(\mathcal{U})$, we must simulate the operation of an α-decision tree for $\mathcal{U}$ that has the minimum depth. Obviously, for the same table $\mathcal{U}$ and different tuples of attribute values, we must simulate the operation of the same decision tree.

3 Auxiliary Statements

In this section, we consider two parameters of distributed decision tables and three statements without proofs regarding these parameters.

We define now a parameter $D(\mathcal{U})$ of the distributed decision table $\mathcal{U}$. If $\mathcal{U}$ is degenerate, then $D(\mathcal{U}) = 0$. Let $\mathcal{U}$ be nondegenerate and $At(\mathcal{U}) = \{g_1, \ldots, g_m\}$. Let $\bar{\sigma} = (\sigma_1, \ldots, \sigma_m) \in E_k^m$. We denote by $D(\mathcal{U}, \bar{\sigma})$ the minimum length of a word β over the alphabet $\{(g_1, \sigma_1), \ldots, (g_m, \sigma_m)\}$ for which $\mathcal{U}\beta$ is degenerate. Then $D(\mathcal{U}) = \max\{D(\mathcal{U}, \bar{\sigma}) : \bar{\sigma} \in E_k^m\}$.

Lemma 1. *Let $\mathcal{U}$ be a distributed decision table and $\mathcal{U}'$ be a subtable of $\mathcal{U}$. Then $D(\mathcal{U}') \leq D(\mathcal{U})$.*

Lemma 2. *Let $\mathcal{U}$ be a distributed decision table. Then $d_0(\mathcal{U}) \geq D(\mathcal{U})$.*

Lemma 3. *Let $\mathcal{U}$ be a distributed decision table, $\mathcal{U}'$ be a subtable of $\mathcal{U}$, $g_i \in At(\mathcal{U})$, and $\sigma \in E_k$. Then $R(\mathcal{U}) - R(\mathcal{U}(g_i, \sigma)) \geq R(\mathcal{U}') - R(\mathcal{U}'(g_i, \sigma))$.*

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

4 Main Results

In this section, we examine the problem of minimizing the depth of an α-decision tree. We show that this problem is NP-hard. As a first step, we focus on analyzing a greedy algorithm for α-decision tree depth minimization.

Let α be a real number with $0 \leq \alpha < 1$. We now describe a greedy algorithm $\mathcal{W}_\alpha$ which, for a given distributed decision table $\mathcal{U} = (U_1, \ldots, U_n)$ with $At(\mathcal{U}) = \{g_1, \ldots, g_m\}$, simulates the operation of an α-decision tree $\mathcal{W}_\alpha(\mathcal{U})$ for $\mathcal{U}$ on a given tuple $\bar{\sigma} = (\sigma_1, \ldots, \sigma_m) \in E_k^m$ of values of the attributes $g_1, \ldots, g_m$.

Algorithm $\mathcal{W}_\alpha$

Step 1. Set $\beta = \lambda$.

Step 2. If $R(\mathcal{U}\beta) \leq \alpha R(\mathcal{U})$, then the decision tree $\mathcal{W}_\alpha(\mathcal{U})$ returns the general decision $(s_1, \ldots, s_n)$ where, for $j = 1, \ldots, n$, $s_j = \emptyset$ if the table $U_j\beta$ is empty and s_j is the most common decision for $U_j\beta$ if this table is nonempty.

If $R(\mathcal{U}\beta) > \alpha R(\mathcal{U})$, then, for $i = 1, \ldots, m$, compute the value

$$Q(g_i) = \max\{R(\mathcal{U}\beta(g_i, \sigma)) : \sigma \in E_k\}$$

and choose the minimum $i \in \{1, \ldots, m\}$ for which $Q(g_i)$ has the minimum value. The decision tree $\mathcal{W}_\alpha(\mathcal{U})$ computes the value of the attribute g_i and obtains that $g_i = \sigma_i$. Set $\beta := \beta(g_i, \sigma_i)$ and go to Step 2.

Note 1. If $\mathcal{U}$ is a degenerate distributed decision table, then $d(\mathcal{W}_\alpha(\mathcal{U})) = 0$.

Theorem 1. *Let $\mathcal{U}$ be a nondegenerate distributed decision table and α be a real number such that $0 < \alpha < 1$. Then $d(\mathcal{W}_\alpha(\mathcal{U})) \leq D(\mathcal{U}) \ln \frac{1}{\alpha} + 1$.*

Proof. Let $At(\mathcal{U}) = \{g_1, \ldots, g_m\}$. For each $i = 1, \ldots, m$, let γ_i denote the smallest element of E_k such that

$$R(\mathcal{U}(g_i, \gamma_i)) = \max\{\, R(\mathcal{U}(g_i, \gamma)) : \gamma \in E_k \,\}.$$

Clearly, $Q(g_i) = R(\mathcal{U}(g_i, \gamma_i))$. It follows that the root of the tree $\mathcal{W}_\alpha(\mathcal{U})$ is labeled with the attribute g_{i_0}, where i_0 is the smallest index i for which $R(\mathcal{U}(g_i, \gamma_i))$ attains its minimum value.

Let us show that

$$R(\mathcal{U}(g_{i_0}, \gamma_{i_0})) \leq (1 - 1/D(\mathcal{U}))\, R(\mathcal{U}).$$

It is clear that there exist attributes $g_{i_1}, \ldots, g_{i_t} \in \{g_1, \ldots, g_m\}$ such that

$$\mathcal{U}(g_{i_1}, \gamma_{i_1}) \ldots (g_{i_t}, \gamma_{i_t})$$

is degenerate and $t \leq D(\mathcal{U})$. Evidently, $R(\mathcal{U}(g_{i_1}, \gamma_{i_1}) \ldots (g_{i_t}, \gamma_{i_t})) = 0$. Therefore

$$\begin{aligned} & R(\mathcal{U}) - [R(\mathcal{U}) - R(\mathcal{U}(g_{i_1}, \gamma_{i_1}))] - [R(\mathcal{U}(g_{i_1}, \gamma_{i_1})) - R(\mathcal{U}(g_{i_1}, \gamma_{i_1})(g_{i_2}, \gamma_{i_2}))] - \\ & \ldots - [R(\mathcal{U}(g_{i_1}, \gamma_{i_1}) \ldots (g_{i_{t-1}}, \gamma_{i_{t-1}})) - R(\mathcal{U}(g_{i_1}, \gamma_{i_1}) \ldots (g_{i_t}, \gamma_{i_t}))] \\ = {} & R(\mathcal{U}(g_{i_1}, \gamma_{i_1}) \ldots (g_{i_t}, \gamma_{i_t})) = 0. \end{aligned}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

From Lemma 3 it follows that, for $j = 1, \dots, t-1$, $R(\mathcal{U}(g_{i_1}, \gamma_{i_1}) \dots (g_{i_j}, \gamma_{i_j})) - R(\mathcal{U}(g_{i_1}, \gamma_{i_1}) \dots (g_{i_j}, \gamma_{i_j})(g_{i_{j+1}}, \gamma_{i_{j+1}})) \le R(\mathcal{U}) - R(\mathcal{U}(g_{i_{j+1}}, \gamma_{i_{j+1}}))$.

Therefore $R(\mathcal{U}) - \sum_{j=1}^{t}(R(\mathcal{U}) - R(\mathcal{U}(g_{i_j}, \gamma_{i_j}))) \le 0$. Since $R(\mathcal{U}(g_{i_0}, \gamma_{i_0})) \le R(\mathcal{U}(g_{i_j}, \gamma_{i_j}))$ for $j = 1, \dots, t$, we have $R(\mathcal{U}) - t(R(\mathcal{U}) - R(\mathcal{U}(g_{i_0}, \gamma_{i_0}))) \le 0$ and $R(\mathcal{U}(g_{i_0}, \gamma_{i_0})) \le (1 - 1/t)R(\mathcal{U})$. Taking into account that $t \le D(\mathcal{U})$, we obtain $R(\mathcal{U}(g_{i_0}, \gamma_{i_0})) \le (1 - 1/D(\mathcal{U}))\, R(\mathcal{U})$.

Assume first that $D(\mathcal{U}) = 1$. From the obtained inequality and from the description of the algorithm $\mathcal{W}_\alpha$, it follows that $d(\mathcal{W}_\alpha(\mathcal{U})) = 1$. Thus, if $D(\mathcal{U}) = 1$, the statement of the theorem holds.

Now let $D(\mathcal{U}) \ge 2$. Consider a longest path in the tree $\mathcal{W}_\alpha(\mathcal{U})$ from the root to a terminal node, and let its length be p. Suppose the working nodes along this path are labeled with the attributes $g_{j_1}, \dots, g_{j_p}$, where $g_{j_1} = g_{i_0}$, and the corresponding edges are labeled with the values $\sigma_1, \dots, \sigma_p$. For each $q = 1, \dots, p$, denote by $\mathcal{U}_q$ the distributed table $\mathcal{U}(g_{j_1}, \sigma_1) \cdots (g_{j_q}, \sigma_q)$. By Lemma 1, we have $D(\mathcal{U}_q) \le D(\mathcal{U})$ for all $q = 1, \dots, p$. Moreover, we have established that

$$R(\mathcal{U}_1) \le R(\mathcal{U})\big(1 - \tfrac{1}{D(\mathcal{U})}\big).$$

One can similarly show that

$$R(\mathcal{U}_q) \le R(\mathcal{U})\big(1 - \tfrac{1}{D(\mathcal{U})}\big)^q, \quad q = 1, \dots, p.$$

Consider the table $\mathcal{U}_{p-1}$. For this table, we have

$$R(\mathcal{U}_{p-1}) \le R(\mathcal{U})\big(1 - \tfrac{1}{D(\mathcal{U})}\big)^{p-1}.$$

From the description of the algorithm $\mathcal{W}_\alpha$, it follows that

$$R(\mathcal{U}_{p-1}) > \alpha R(\mathcal{U}).$$

Hence,

$$\alpha < \big(1 - \tfrac{1}{D(\mathcal{U})}\big)^{p-1} \quad \text{and} \quad \big(1 + \tfrac{1}{D(\mathcal{U})-1}\big)^{p-1} < \frac{1}{\alpha}.$$

Taking the natural logarithm of both sides gives

$$(p-1)\ln\Big(1 + \frac{1}{D(\mathcal{U}) - 1}\Big) \le \ln\frac{1}{\alpha}.$$

It is known that for any natural number r,

$$\ln(1 + \tfrac{1}{r}) > \frac{1}{r+1}.$$

Since $D(\mathcal{U}) \ge 2$, we obtain

$$\frac{p-1}{D(\mathcal{U})} < \ln\frac{1}{\alpha} \quad \text{and hence} \quad p < D(\mathcal{U})\ln\frac{1}{\alpha} + 1.$$

Taking into account that $p = d(\mathcal{W}_\alpha(\mathcal{U}))$, we conclude that

$$d(\mathcal{W}_\alpha(\mathcal{U})) < D(\mathcal{U})\ln\frac{1}{\alpha} + 1.$$

□

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

Using Theorem 1 and Lemma 2 we obtain the following

Corollary 1. *For any nondegenerate distributed decision table* $\mathcal{U}$ *and any real number* α, $0 < \alpha < 1$, $d(\mathcal{W}_\alpha(\mathcal{U})) \leq d_0(\mathcal{U}) \ln \frac{1}{\alpha} + 1$.

Based on the results obtained in [8] we can prove the following statement.

Proposition 1. *For any* α *with* $0 \leq \alpha < 1$*, the problem of minimizing the depth of an* α*-decision tree is* NP*-hard.*

5 Recognizing of Point Color

In this section, we study distributed decision table $\mathcal{U} = (U_1, \ldots, U_n)$ in which each decision table is associated with the problem of recognizing the color of a point from a finite set of two-colored points in the plane. To solve this problem, we use attributes corresponding to vertical and horizontal straight lines. We show that $D(\mathcal{U}) \leq 4$ and the depth of shared α-decision trees constructed by the greedy algorithm $\mathcal{W}_\alpha$ is bounded above by $4 \ln \frac{1}{\alpha} + 1$.

Let we have $m+1$, $m \geq 2$, vertical straight lines in the plane given by equations $x = 1, \ldots, x = m+1$ and $m+1$ horizontal straight lines given by equations $y = 1, \ldots, y = m+1$. These lines form m^2 squares. Let we have a set C of q points, $2 \leq q \leq m^2$, inside of these squares such that in one square we can have at most one point. Let these points are colored into two colors, white and green, such that at least one point is green and at least one point is white. We should recognize the color of a given point using the attributes $g_1, \ldots, g_{2m+2}$ with values from the set E_2. For a point (a, b), the value of the attribute g_i is defined in the following way. If $i \in \{1, \ldots, m+1\}$, then $g_i(x, y) = 0$ if and only if $a < i$. If $i \in \{m+2, \ldots, 2m+2\}$, then $g_i(x, y) = 0$ if and only if $b < i - m - 1$.

We correspond to the considered problem a 2-valued decision table U with $2m+2$ columns labeled with the attributes $g_1, \ldots, g_{2m+2}$ and q rows corresponding to the points from C. Let $C = \{p_1, \ldots, p_q\}$. For $j = 1, \ldots, q$, the jth row of U contains values of the attributes $g_1, \ldots, g_{2m+2}$ on the point p_j. This row is labeled with the decision 1 if p_j is a white point and with the decision 2 if p_j is a green point. We will say about the table U as about *m-RPC-table*. We avoid the proof of the following statement.

Lemma 4. *Let* $m \geq 2$ *and* $\mathcal{U} = (U_1, \ldots, U_n)$ *be a distributed decision table in which, for* $i = 1, \ldots, n$, U_i *is an m-RPC-table. Then* $D(\mathcal{U}) \leq 4$.

Using Theorem 1 and Lemma 4, we obtain the following

Corollary 2. *Let* $0 < \alpha < 1$, $m \geq 2$ *and* $\mathcal{U} = (U_1, \ldots, U_n)$ *be a distributed decision table in which, for* $i = 1, \ldots, n$, U_i *is an m-RPC-table. Then* $d(\mathcal{W}_\alpha(\mathcal{U})) \leq 4 \ln \frac{1}{\alpha} + 1$.

Interestingly, the upper bound under consideration depends only on α.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26

6 Conclusions

In the paper, an approach focused on the analysis of distributed data represented as a tuple of n decision tables was proposed. In this case, object-specific information is distributed across multiple local data sources. Each table may contain a different subset of objects, attributes, introducing challenges related to knowledge integration and heterogeneous data structures. Since constructing an approximate decision tree with the minimum depth that can be applied simultaneously to each table (a shared decision tree) is an NP-hard problem, a greedy algorithm was proposed. Noteworthy theoretical results including bounds for the depth of decision trees constructed by this algorithm were obtained. In future work, computer experiments to validate the theoretical findings and examine the practical effectiveness of the proposed approach will be performed.

Acknowledgments. Research reported in this publication was supported by the Remote Working Individual Consultancy Agreement No. KAUST-2025-C0148, and King Abdullah University of Science and Technology (KAUST) and the Institute of Computer Science, University of Silesia in Katowice, Poland.

References

1. Buhrman, H., de Wolf, R.: Complexity measures and decision tree complexity: a survey. Theor. Comput. Sci. **288**(1), 21–43 (2002)
2. da Silva, J.C., Giannella, C., Bhargava, R., Kargupta, H., Klusch, M.: Distributed data mining and agents. Engineering Applications of Artificial Intelligence **18**(7), 791–807 (2005)
3. Grigoriev, D., Karpinski, M., Yao, A.C.: An exponential lower bound on the size of algebraic decision trees for Max. Computational Complexity **7**(3), 193–203 (1998)
4. Loyola-González, O., Ramírez-Sáyago, E., Medina-Pérez, M.A.: Towards improving decision tree induction by combining split evaluation measures. Knowledge-Based Systems **277**, 110832 (2023)
5. Moshkov, M.: Comparative Analysis of Deterministic and Nondeterministic Decision Trees, Intelligent Systems Reference Library, vol. 179. Springer (2020)
6. Moshkov, M.: Time and space complexity of deterministic and nondeterministic decision trees. Ann. Math. Artif. Intell. **91**(1), 45–74 (2023)
7. Moshkov, M., Zielosko, B.: Combinatorial Machine Learning - A Rough Set Approach, Studies in Computational Intelligence, vol. 360. Springer (2011)
8. Slezak, D.: Normalized decision functions and measures for inconsistent decision tables analysis. Fundam. Informaticae **44**(3), 291–319 (2000)
9. Yao, A.C.: Decision tree complexity and Betti numbers. In: 26th Annual ACM Symposium on Theory of Computing, STOC 1994. pp. 615–624 (1994)
10. Zielosko, B., Moshkov, M., Tetteh, E.T.: Optimization of inner and general rules. Inf. Sci. **719**, 122466 (2025)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_26