‘Misclassification Error’ Greedy Heuristic to Construct Decision Trees for Inconsistent Decision Tables

Mohammad Azad and Mikhail Moshkov
Computer, Electrical & Mathematical Sciences & Engineering Division,
King Abdullah University of Science and Technology,
Thuwal 23955-6900, Saudi Arabia

Keywords: Optimization, Decision Trees, Dynamic Programming, Greedy Heuristics, Many-valued Decisions.

Abstract: A greedy algorithm has been presented in this paper to construct decision trees for three different approaches (many-valued decision, most common decision, and generalized decision) in order to handle the inconsistency of multiple decisions in a decision table. In this algorithm, a greedy heuristic ‘misclassification error’ is used which performs faster, and for some cost function, results are better than ‘number of boundary subtables’ heuristic in literature. Therefore, it can be used in the case of larger data sets and does not require huge amount of memory. Experimental results of depth, average depth and number of nodes of decision trees constructed by this algorithm are compared in the framework of each of the three approaches.

1 INTRODUCTION

There are groups of rows (objects) with equal values of conditional attributes but with different decisions (values of the decision attribute) in inconsistent decision tables (IDT). It is common to have such tables in our real life because we do not have enough number of attributes of the domain to separate rows. Furthermore, it is natural to have such data sets in optimization problems such as finding a Hamiltonian circuit with the minimum length in traveling salesman problem, finding nearest post office (Moshkov and Zielosko, 2011). It also arises when we study, e.g., problem of semantic annotation of images (Boutell et al., 2004), music categorization into emotions (Wieczorkowska et al., 2005), functional genomics (Blockeel et al., 2006), text categorization (Zhou et al., 2005) etc.

Table 1 presents the ‘Play Tennis’ example (Mitchell, 1997) where the conditional attributes describe the condition of the environment and the decision attribute refers whether one can play tennis or not (Note, r_i refers observation or row i). Here, r_1, r_8 and r_{15} have the same values of conditional attributes but different decisions. Such rows are highlighted in blue color. Similar situation is for r_6, and r_{10} that have been colored with red. This type of inconsistency can happen because of missing attribute to separate objects. In the paper (Azad et al., 2013), three approaches are considered to deal with inconsistent decision tables.

Table 1: Example of ‘Play Tennis’ inconsistent decision table.

	Outlook	Humidity	Wind	Play Tennis
r_1	Sunny	High	Weak	No
r_2	Sunny	High	Strong	No
r_3	Overcast	High	Weak	Yes
r_4	Rain	High	Weak	Yes
r_5	Rain	Normal	Weak	Yes
r_6	Rain	Normal	Strong	No
r_7	Overcast	Normal	Weak	Yes
r_8	Sunny	High	Weak	Yes
r_9	Sunny	Normal	Weak	Yes
r_{10}	Rain	Normal	Strong	Yes
r_{11}	Sunny	Normal	Strong	Yes
r_{12}	Overcast	High	Strong	Yes
r_{13}	Overcast	Normal	Weak	Yes
r_{14}	Rain	High	Strong	No
r_{15}	Sunny	High	Weak	Yes

The first approach is called many-valued decisions – *MVD*. Instead of a group of equal rows with different decisions, just one row is kept with the same values of conditional attributes and a set containing all decisions for rows from the group (Moshkov and Zielosko, 2011) is attached with the row. The second approach is called the most common decision – *MCD*. Instead of a group of equal rows with different deci-

Azad M. and Moshkov M..
‘Misclassification Error’ Greedy Heuristic to Construct Decision Trees for Inconsistent Decision Tables.
DOI: 10.5220/0005059201840191
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2014), pages 184-191
ISBN: 978-989-758-048-2

sions, just one row is kept with the same values of conditional attributes and a single decision attached with this row that have the value of the most common decision for rows from the group. The third approach is well known in the rough set theory (Pawlak, 1991; Skowron and Rauszer, 1992) and is called generalized decision – *GD* approach. In this case, an inconsistent decision table is transformed into the table with many-valued decisions and after that each set of decisions is encoded by a number (decision) such that equal sets are encoded by equal numbers and different sets are encoded by different numbers.

In literature, often, problems that are connected with multi-label data are considered for classification (multi-label classification problem) (Clare and King, 2001; Comité et al., 2003; Loza Mencía and Fürnkranz, 2008; Tsoumakas and Katakis, 2007; Tsoumakas et al., 2010; Zhou et al., 2012). However, in this paper the aim is to study decision trees for multi-label decision tables for knowledge representation. Furthermore, decision tree has been used as the model to represent such knowledge and the goal is to compare the decision tree structure for different heuristics and different approaches of representation of inconsistent decision table.

In this paper, a greedy heuristic 'misclassification error' has been introduced for decision table with many-valued decisions. Its performance has been compared with the heuristic 'number of boundary subtables' from the paper (Azad et al., 2013). Data sets have been used from UCI ML repository as well as KEEL repository (Alcal-Fdez et al., 2009). The advantages of the KEEL data sets are twofolds: (1) they have big amount of data, and (2) they are real life examples of decision tables with many-valued decisions. The heuristic 'number of boundary subtables' is complex in terms of time, and memory requirement. Hence, it is essential to design new heuristic which gives equal or better results but with less time complexity, and memory requirement. At the end, results have been presented which show that the use of *MCD* and, especially, *MVD* approaches can reduce the complexity of trees in comparison with *GD* approach. The goal is to find all decisions for the case *GD* whereas a fixed decision for the case of *MCD* and arbitrary decision for the case of *MVD* for a particular row. That means we are moving from highly restricted decision constraint to less restricted decision constraint. Hence we usually get less complex tree for the last case than others. This comparison is crucial for knowledge representation since we can get the useful knowledge in the form of less complicated decision trees.

This paper consists of five sections. Section 2 contains main definitions. In Sect. 3, the greedy algorithm for construction of decision trees is presented. Section 4 contains results of experiments and Sect. 5 concludes the paper.

2 MAIN DEFINITIONS

A *decision table* is a rectangular table T filled by non-negative integers. Columns of this table are labeled with conditional attributes $f_1, \ldots, f_n$. If we have strings as values of attributes, we have to encode the values as nonnegative integers. In addition, if we have real valued data, we have to discretize the value to use in this format. Rows of the table are pairwise different, and each row is labeled with a natural number (decision) which is interpreted as a value of the decision attribute. To differentiate with *decision table with many-valued decisions*, we sometimes call it *decision table with one-valued decisions*.

It is possible that T is inconsistent, i.e., contains equal rows with different decisions. The table T can contain also equal rows with equal decisions. The most frequent decision attached to rows from a group of rows in a decision table T with one-valued decision is called the *most common decision* for this group of rows. For approach called *most common decision – MCD*, we transform inconsistent decision table T into consistent decision table T_{MCD} with one-valued decision. Instead of a group of equal rows with different decisions, we consider one row from the group and we attach to this row the most common decision for the considered group of rows.

For approach called *generalized decision – GD*, we transform inconsistent decision table T into consistent decision table T_{GD} with one-valued decisions. Instead of a group of equal rows with different decisions, we consider one row from the group and we attach to this row the set of all decisions for rows from the group. Then instead of a set of decisions we attach to each row a code of this set – a natural number such that the codes of equal sets are equal and the codes of different sets are different.

For approach called *many-valued decisions – MVD*, we transform an inconsistent decision table T into a decision table T_{MVD} with many-valued decisions. Instead of a group of equal rows with different decisions, we consider one row from the group and we attach to this row the set of all decisions for rows from the group (Moshkov and Zielosko, 2011).

Note that each decision table with one-valued decisions can be interpreted also as a decision table with many-valued decisions. In such table, each row is labeled with a set of decisions which has one element.

Table 2: Transformation of inconsistent decision table T^0 into decision tables T^0_{MVD}, T^0_{GD} and T^0_{MCD}.

$T^0 =$

	f_1	f_2	f_3	
r_1	0	0	0	1
r_2	0	1	1	1
r_3	0	1	1	2
r_4	1	0	1	1
r_5	1	0	1	3
r_6	1	1	0	2
r_7	1	1	0	3
r_8	0	0	1	2

$T^0_{MVD} =$

	f_1	f_2	f_3	
r_1	0	0	0	{1}
r_2	0	1	1	{1,2}
r_3	1	0	1	{1,3}
r_4	1	1	0	{2,3}
r_5	0	0	1	{2}

$T^0_{GD} =$

	f_1	f_2	f_3	
r_1	0	0	0	1
r_2	0	1	1	2
r_3	1	0	1	3
r_4	1	1	0	4
r_5	0	0	1	5

$T^0_{MCD} =$

	f_1	f_2	f_3	
r_1	0	0	0	1
r_2	0	1	1	1
r_3	1	0	1	1
r_4	1	1	0	2
r_5	0	0	1	2

We have shown in Table 2 the transformation of an inconsistent decision table T^0 using all the three approaches.

We denote row i by r_i where $i = 1,\ldots,N(T)$. For example, r_1 means the first row, r_2 means the second row and so on. We denote the number of rows in the table T by $N(T)$.

If there is a decision which belongs to the set of decisions attached to each row of T, then we call it a *common decision* for T. We will say that T is a *degenerate* table if T does not have rows or it has a common decision. For example, T'_{MVD} is degenerate table as shown in Table 3, where the common decision is 1.

Table 3: A degenerate many-valued decision table, T'.

$T'_{MVD} =$

	f_1	f_2	f_3	
r_1	0	0	0	{1}
r_2	0	1	1	{1,2}
r_3	1	0	1	{1,3}

A table obtained from T by removing some rows is called a subtable of T. There is a special type of subtable called *boundary subtable*. The subtable T' of T is a *boundary subtable* of T if and only if T' is not degenerate but each of its proper subtable is degenerate. We denote the number of boundary subtables of the table T by $nBS(T)$. It is clear that T is a degenerate table if and only if $nBS(T) = 0$. The value $nBS(T)$ has been considered as greedy heuristic of T in (Azad et al., 2013). Below is an example of all boundary subtables of T^0_{MVD}:

$T_1 =$

	f_1	f_2	f_3	d
r_2	0	1	1	{1,2}
r_3	1	0	1	{1,3}
r_4	1	1	0	{2,3}

$T_2 =$

	f_1	f_2	f_3	d
r_1	0	0	0	{1}
r_4	1	1	0	{2,3}

$T_3 =$

	f_1	f_2	f_3	d
r_3	1	0	1	{1,3}
r_5	0	0	1	{2}

$T_4 =$

	f_1	f_2	f_3	d
r_1	0	0	0	{1}
r_5	0	0	1	{2}

We denote the subtable of T which consists of rows that have values $a_1,\ldots,a_m$ at the intersection with columns $f_{i_1},\ldots,f_{i_m}$ by $T(f_{i_1},a_1)\ldots(f_{i_m},a_m)$. Such nonempty subtables (including the table T) are called separable subtables of T. For example, (see Table 4) if we consider subtable $T^0_{MVD}(f_1,0)$ for table T^0_{MVD} (see Table 2), it will consist of rows $1,2$, and 5. Similarly, $T^0_{MVD}(f_1,0)(f_2,0)$ subtable will consist of rows 1, and 5.

Table 4: Example of subtables of a decision table with many-valued decisions T^0_{MVD}.

$T^0_{MVD}(f_1,0) =$

	f_1	f_2	f_3	
r_1	0	0	0	{1}
r_2	0	1	1	{1,2}
r_5	0	0	1	{2}

$T^0_{MVD}(f_1,0)(f_2,0) =$

	f_1	f_2	f_3	
r_1	0	0	0	{1}
r_5	0	0	1	{2}

The set of attributes (columns of table T) which have different values i.e. not constant, is denoted by $E(T)$. For example, for the table T^0_{MVD}, $E(T^0_{MVD}) = \{f_1,f_2,f_3\}$. Similarly, $E(T^0_{MVD}(f_1,0)) = \{f_2,f_3\}$ for the subtable $T^0_{MVD}(f_1,0)$, because the value for the attribute f_1 is constant (=0) in subtable $T^0_{MVD}(f_1,0)$. The set of values from the column f_i is denoted by $E(T,f_i)$ where $f_i \in E(T)$. For example, if we consider table T^0_{MVD} and attribute f_1, then $E(T^0_{MVD},f_1) = \{0,1\}$.

The *most common decision* for a table T with many-valued decisions is the decision which belongs to the maximum number of sets of decisions attached to rows of the table T. If we have more than one such decision, we choose the minimum one. We denote the number of rows for which the set of decisions contains the most common decision by $N_{mcd}(T)$. We denote the *Misclassification error* for a table T which is the difference between total number of rows and number of rows with the most common decision in the set of decisions in the table T by $M(T)$, i.e., $M(T) =$

$N(T) - N_{mcd}(T)$. For example, if we look at the table T^0_{MVD}, the decisions $1,2,3$, appear $3,3,2$ times, respectively. The minimum decision that appears most of the time in the table T^0_{MVD} is 1. Therefore, 1 is the most common decision in T^0_{MVD}. So, for the table T^0_{MVD}, the number of rows $N(T^0_{MVD})$ is 5, number of rows with most common decision $N_{mcd}(T^0_{MVD})$ is 3, and misclassification error is $5-3=2$.

A *decision tree over* T is a finite tree with root in which each terminal node is labeled with a decision (a natural number), and each nonterminal node is labeled with an attribute from the set $\{f_1,\dots,f_n\}$. A number of edges start from each nonterminal node which are labeled with different non-negative integers (e.g. two edges labeled with 0 and 1 if the nonterminal node is labeled with binary attribute).

Let Γ be a decision tree over T and v be a node of Γ. There is one to one mapping between node v and subtable of T i.e. for each node v, there is a unique subtable of T. This subtable is defined as $T(v)$ corresponding to the table T and node v. If node v is the root of Γ then $T(v) = T$ i.e. the subtable $T(v)$ is the same as T. Otherwise, $T(v)$ is the subtable $T(f_{i_1},\delta_1)\dots(f_{i_m},\delta_m)$ of the table T where attributes $f_{i_1},\dots,f_{i_m}$ and numbers $\delta_1,\dots,\delta_m$ are respectively node and edge labels in the path from the root to node v.

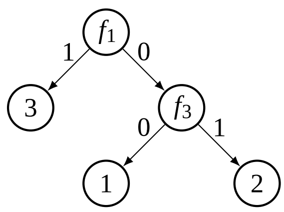

Figure 1: Decision tree for T^0_{MVD}.

We will say that Γ is a decision tree for T, if for any node v of Γ:

- if $T(v)$ is degenerate then v is labeled with the common decision for $T(v)$,
- if $T(v)$ is not degenerate then v is labeled with an attribute $f_i \in E(T(v))$, and if $E(T(v), f_i) = \{a_1,\dots,a_k\}$, then k outgoing edges from node v are labeled with $a_1,\dots,a_k$.

An example of a decision tree for the table T^0_{MVD} can be found in Fig. 1. If v is the node labeled with the attribute f_3, then subtable $T(v)$ corresponding to the node v will be the subtable $T(f_1,0)$ of table T. Similarly, the subtable corresponding to the node labeled with 2 will be $T(f_1,0)(f_3,1)$.

The *depth* of Γ which is the maximum length of a path from the root to a terminal node is denoted by $h(\Gamma)$. Let $\Delta(T)$ be the set of rows of T. The average depth of Γ is denoted by $h_{avg}(\Gamma)$ which is equal to $\sum_{r\in\Delta(T)} \frac{l_\Gamma(r)}{N(T)}$, where $l_\Gamma(r)$ is the length of the path from the root of Γ to a terminal node v for which r belongs to $T(v)$. The number of nodes in the decision tree Γ is denoted by $L(\Gamma)$.

Algorithm 1: Greedy algorithm U.

Input: A decision table T with many-valued decisions, and conditional attributes $f_1,\dots,f_n$.
Output: Decision tree $U(T)$ for T.

Construct the tree G consisting of a single node labeled with the table T;
while (true) **do**
  if No node of the tree G is labeled with a table **then**
    Denote the tree G by $U(T)$;
  else
    Choose a node v in G which is labeled with a subtable T' of the table T;
    if $M(T') = 0$ **then**
      Instead of T' mark the node v with the most common decision for T';
    else
      For each $f_i \in E(T')$, we compute the value of the impurity function $I(T', f_i)$ equal to $\sum_{b\in E(T',f_i)} M(T'(f_i,b)) \times N(T'(f_i,b))$;
      Choose the attribute $f_{i_0} \in E(T')$, where i_0 is the minimum i for which $I(T', f_i)$ has the minimum value; Instead of T' mark the node v with the attribute f_{i_0};
      For each $\delta \in E(T', f_i)$, add to the tree G the node v_δ and mark this node with the subtable $T'(f_{i_0},\delta)$;
      Draw an edge from v to v_δ and mark this edge with δ;
    end if
  end if
end while

3 GREEDY ALGORITHM U

The greedy algorithm U, for a given decision table T with many-valued decisions, constructs a decision tree $U(T)$ for T (see Algorithm 1). We will interpret decision table with one-valued decisions, i.e. T_{GD}, T_{MCD}, as a decision table with many-valued decisions where each row is labeled with a set of decisions that has one element. Hence, we can apply the same algorithm for all three cases.

In Algorithm 1, the heuristic M refers the misclassification error for the given decision table T. The

Table 5: Characteristics of decision tables with many-valued decisions from KEEL data sets.

Decision table T	Rows	Attr	Spectrum									
			#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
*bibtex**	7355	1836	2791	1825	1302	669	399	179	87	46	18	7
corel5k	4998	499	3	376	1559	3013	17	0	1	0	0	0
*delicious**	15862	944	95	207	292	340	422	536	714	930	1108	1460
*enron**	1561	1001	179	238	441	337	200	91	51	15	3	3
genbase	662	1186	560	58	31	8	2	3	0	0	0	0
medical	967	1449	741	212	14	0	0	0	0	0	0	0

number of rows $N(T'(f_i,b))$ is used as the weight of the corresponding table $T'(f_i,b)$. The impurity function I has been calculated as the weighted sum of the values of the heuristic, i.e., the M value is multiplied with the weight of the corresponding table. Then, the multiplied values are summed up.

Now, the work of the greedy algorithm for construction of a decision tree for the decision table T^0_{MVD} is portrayed in Fig. 1. The table T^0_{MVD} is not degenerate, so for $i \in \{1,2,3\}$, the value of $I(f_i)$ has been computed: $I(f_1) = 0+3 = 3$, $I(f_2) = 0+3 = 3$, $I(f_3) = 3+2 = 5$. The minimum values are $I(f_1)$, and $I(f_2)$. Then, the attribute f_1 is selected as it is with minimum index. After that, it is assigned to the root of the constructed tree.

The left child of the root denotes the node which points the subtable $T^0_{MVD}(f_1,1)$ and the edge is marked by 1. The subtable $T^0_{MVD}(f_1,1)$ is degenerate and the common decision is 3. So, the node is labeled by the decision 3. Similarly, the right child of the root denote the subtable $T^0_{MVD}(f_1,0)$ which is not degenerate. Then, it is further divided according to the algorithm 1 by choosing the attribute f_3. The child nodes of the node labeled with f_3 are degenerate and they are marked by their common decisions. As all nodes are labeled, the work of the algorithm is finished. At the end, the decision tree is $U(T^0_{MVD})$ (see Fig. 1) for the table T^0_{MVD}.

4 EXPERIMENTAL RESULTS

A number of decision tables T from KEEL multi-label data sets (Alcal-Fdez et al., 2009) as well as from UCI ML repository (Bache and Lichman, 2013) have been considered. Data sets from KEEL are already in decision table with many-valued decisions format T_{MVD}. These tables T are further converted into formats T_{MCD} (in this case, the first decision is selected from the set of decisions attached to a row) and T_{GD} by the procedure described in Section 2. Conversely, more conditional attributes are removed from the data sets of UCI ML Repository to convert data sets into inconsistent decision tables. These inconsistent tables were further converted into many-valued, most common and generalized decision by the procedure described in Section 2. The information about the considered decision tables is shown in Table 5 and 7.

These tables contain name of decision table $T = T_{MVD}$, number of rows (column "Rows"), number of attributes (column "Attr"), and spectrum of the table T (column "Spectrum"). Spectrum of a decision table with many-valued decisions is a sequence #1, #2,…, where #i, $i = 1,2,\ldots$, is the number of rows labeled with sets of decisions with the cardinality equal to i. For some tables (marked with * in Table 5), the spectrum is too long to fit in the page width. Hence, it has been shown up to the element of the sequence that is possible to show in the page width limit.

Table 6 and 9 contain depth, average depth and number of nodes for decision trees $U(T_{MVD})$, $U(T_{MCD})$ and $U(T_{GD})$ constructed by the greedy algorithm U using greedy heuristic 'misclassification error' for decision tables T_{MVD}, T_{MCD} and T_{GD} respectively. For the depth of the trees, even though some cases ('enron', 'genbase') *MVD* approach gives larger tree depth, but on average it gives smallest depth of the tree. For other cost functions like average depth and number of nodes, it gives equal or smaller tree size than *MCD* and *GD* approach. Now, if one compares between *MCD* approach and *GD* approach, one can find that *GD* approach gives either equal or larger tree size.

Therefore, the decision trees constructed in the frameworks of *MVD* approach are usually simpler than the decision trees constructed in the frameworks of *MCD* approach, and the decision trees constructed in the frameworks of *MCD* approach are usually simpler than the decision trees constructed in the framework of *GD* approach.

Furthermore, the decision tree depth, average depth and number of nodes has been compared based on greedy heuristics. In the paper (Azad et al., 2013), the number of boundary subtables has been considered as heuristic to compare among three approaches.

Table 6: Depth, average depth and number of nodes for decision trees $U(T_{MVD})$, $U(T_{GD})$ and $U(T_{MCD})$.

Decision table T	Depth *MVD*	*MCD*	*GD*	Average Depth *MVD*	*MCD*	*GD*	Number of Nodes *MVD*	*MCD*	*GD*
bibtex	39	42	43	11.52	12.24	12.97	9357	10583	13521
corel5k	156	156	157	36.1	36.41	36.29	6899	8235	9823
delicious	79	92	92	13.74	15.9	16.718	6455	18463	31531
enron	28	26	41	9.18	9.62	11.18	743	1071	2667
genbase	12	12	11	4.718	4.937	5.762	43	49	81
medical	16	16	16	8.424	8.424	8.424	747	747	747
average	**55**	**57.33**	**60**	**13.95**	**14.59**	**15.22**	**4040.67**	**6524.67**	**9728.33**

Table 7: Characteristics of decision tables with many-valued decisions from UCI data sets.

Decision table T	Rows	Attr	Spectrum #1	#2	#3	#4	#5	#6
balance-scale-1	125	3	45	50	30			
breast-cancer-1	193	8	169	24				
breast-cancer-5	98	4	58	40				
cars-1	432	5	258	161	13			
flags-5	171	21	159	12				
hayes-roth-data-1	39	3	22	13	4			
kr-vs-kp-5	1987	31	1564	423				
kr-vs-kp-4	2061	32	1652	409				
lymphography-5	122	13	113	9				
mushroom-5	4078	17	4048	30				
nursery-1	4320	7	2858	1460	2			
nursery-4	240	4	97	96	47			
spect-test-1	164	21	161	3				
teeth-1	22	7	12	10				
teeth-5	14	3	6	3	0	5	0	2
tic-tac-toe-4	231	5	102	129				
tic-tac-toe-3	449	6	300	149				
zoo-data-5	42	11	36	6				

But in this paper, the result using 'misclassification error' heuristic has been compared with the result using 'number of boundary subtables' heuristic. Table 8 shows the results using 'number of boundary subtables' heuristic along with 'weighted-sum' impurity function. If one looks at the Table 8 and 9, the performance of the heuristic 'misclassification error' is comparable to the heuristic 'number of boundary subtable'. In the case of 'depth', it is slightly smaller for 'number of boundary subtables', and in the case of 'average depth', it is slightly smaller for 'misclassification error' heuristic. But the 'number of nodes' of tree is essentially smaller for the 'misclassification error' heuristic.

The reason of using UCI ML repository data sets to compare the two heuristics is that the 'number of boundary subtables' heuristic requires a higher degree polynomial (if one consider the maximum number of decisions in the table is bounded) running time and huge amount of memory. Therefore, smaller data sets from UCI ML repository have been used to compare between the two heuristics. The advantage is that the 'misclassification error' heuristic is simple, and requires little time and few memory as well as it gives better results for the minimization of average depth and number of nodes of the constructed tree.

5 CONCLUSIONS

In this paper, three approaches have been described to represent the useful knowledge from inconsistent decision tables in terms of decision tree model. The complexity of decision trees (depth, average depth, and number of nodes) constructed by the greedy algorithm with 'weighted sum' impurity type and 'misclassification error' heuristic have been compared for these approaches.

Experimental results show that the approach based on the many-valued decisions outperforms the approaches based on the generalized decisions and the most common decisions, also it is found that the performance of new heuristic is sometimes better than the 'number of boundary subtables' heuristic.

In future to get a good decision tree model, we want to investigate the effect of new impurity types and new heuristics for inconsistent decision tables. We also would like to investigate the effect of such algorithms when we consider the prediction problem for inconsistent decision tables.

ACKNOWLEDGEMENT

Research reported in this publication was supported by the King Abdullah University of Science and Technology (KAUST).

Table 8: Depth, average depth and number of nodes for decision trees $U(T_{MVD})$, $U(T_{GD})$ and $U(T_{MCD})$ for UCI data sets using number of boundary subtables heuristic.

Decision	Depth			Average Depth			Number of Nodes		
table T	*MVD*	*MCD*	*GD*	*MVD*	*MCD*	*GD*	*MVD*	*MCD*	*GD*
balance-scale-1	2	3	3	2	2.52	3	31	96	156
breast-cancer-1	6	6	7	3.575	3.658	4.104	150	161	220
breast-cancer-5	3	4	4	1.837	2.184	2.602	49	77	102
cars-1	5	5	5	2.361	3.167	4.007	97	197	312
flags-5	6	6	6	3.772	3.819	3.854	211	219	226
hayes-roth-data-1	2	3	3	1.744	1.974	2.308	17	26	39
kr-vs-kp-5	12	15	15	7.939	8.169	9.08	681	957	1635
kr-vs-kp-4	13	14	15	8.009	8.271	9.18	711	1011	1723
lymphography-5	6	6	7	3.787	4.033	4.189	80	98	110
mushroom-5	6	7	8	2.753	2.768	2.78	219	235	261
nursery-1	7	7	7	2.172	3.48	4.132	220	920	1477
nursery-4	2	4	4	1.333	2.283	2.417	9	53	61
spect-test-1	6	9	9	3.037	3.238	3.482	31	45	53
teeth-1	4	4	4	2.818	2.818	2.818	35	35	35
teeth-5	3	3	3	2.214	2.214	2.214	20	20	20
tic-tac-toe-4	5	5	5	2.957	4.017	4.506	73	174	243
tic-tac-toe-3	6	6	6	4.058	4.577	5.343	191	320	542
zoo-data-5	5	6	7	3.071	3.31	3.952	19	25	41
average	5.5	6.28	6.56	3.3	3.69	4.11	158	259.39	403.11

Table 9: Depth, average depth and number of nodes for decision trees $U(T_{MVD})$, $U(T_{GD})$ and $U(T_{MCD})$ for UCI data sets using misclassification error heuristic.

Decision	Depth			Average Depth			Number of Nodes		
table T	*MVD*	*MCD*	*GD*	*MVD*	*MCD*	*GD*	*MVD*	*MCD*	*GD*
balance-scale-1	2	3	3	2	2.52	3	31	96	156
breast-cancer-1	6	6	7	3.679	3.736	4.104	152	160	217
breast-cancer-5	3	4	4	1.816	2.184	2.602	46	77	102
cars-1	5	5	5	1.958	2.583	3.813	43	101	280
flags-5	6	6	6	3.754	3.801	3.836	210	216	223
hayes-roth-data-1	2	3	3	1.744	1.974	2.308	17	26	39
kr-vs-kp-5	13	14	15	7.802	8.329	9.503	543	873	1539
kr-vs-kp-4	14	14	14	7.87	8.504	9.477	555	915	1635
lymphography-5	7	7	7	3.787	4.115	4.311	77	94	112
mushroom-5	7	7	8	2.772	2.781	2.795	246	253	265
nursery-1	7	7	7	2.169	3.469	4.127	198	832	1433
nursery-4	2	4	4	1.333	2.283	2.083	9	53	54
spect-test-1	6	10	10	3.134	3.274	3.543	35	43	53
teeth-1	4	4	4	2.818	2.818	2.818	35	35	35
teeth-5	3	3	3	2.214	2.214	2.214	20	20	20
tic-tac-toe-4	5	5	5	2.965	4.139	4.286	76	182	216
tic-tac-toe-3	6	6	6	4.145	4.78	5.207	199	362	490
zoo-data-5	4	7	7	3.214	3.714	4.119	19	25	41
average	5.67	6.39	6.56	3.29	3.73	4.12	139.5	242.39	383.89

REFERENCES

Alcal-Fdez, J., Snchez, L., Garca, S., Jesus, M., Ventura, S., Garrell, J., Otero, J., Romero, C., Bacardit, J., Rivas, V., Fernndez, J., and Herrera, F. (2009). KEEL Multi Label Data Sets.

Azad, M., Chikalov, I., and Moshkov, M. (2013). Three approaches to deal with inconsistent decision tables - comparison of decision tree complexity. In *RSFDGrC*, pages 46–54.

Bache, K. and Lichman, M. (2013). UCI machine learning repository.

Blockeel, H., Schietgat, L., Struyf, J., Dzeroski, S., and Clare, A. (2006). Decision trees for hierarchical multilabel classification: A case study in functional genomics. In Fürnkranz, J., Scheffer, T., and Spiliopoulou, M., editors, *PKDD 2006, Berlin, Germany, Proceedings*, volume 4213 of *LNCS*, pages 18–29. Springer.

Boutell, M. R., Luo, J., Shen, X., and Brown, C. M. (2004). Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771.

Clare, A. and King, R. D. (2001). Knowledge discovery in multi-label phenotype data. In *PKDD*, pages 42–53.

Comité, F. D., Gilleron, R., and Tommasi, M. (2003). Learning multi-label alternating decision trees from texts and data. In *MLDM*, pages 35–49.

Loza Mencía, E. and Fürnkranz, J. (2008). Pairwise learning of multilabel classifications with perceptrons. In *IJCNN*, pages 2899–2906.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, Inc., NY, USA, 1 edition.

Moshkov, M. and Zielosko, B. (2011). *Combinatorial Machine Learning – A Rough Set Approach*, volume 360 of *Studies in Computational Intelligence*. Springer.

Pawlak, Z. (1991). *Rough Sets – Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht.

Skowron, A. and Rauszer, C. (1992). The discernibility matrices and functions in information systems. In *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Set Theory*, pages 331–362. Kluwer Academic Publishers, Dordrecht.

Tsoumakas, G. and Katakis, I. (2007). Multi-label classification: An overview. *IJDWM*, 3(3):1–13.

Tsoumakas, G., Katakis, I., and Vlahavas, I. P. (2010). Mining multi-label data. In *Data Mining and Knowledge Discovery Handbook, 2nd ed.*, pages 667–685. Springer.

Wieczorkowska, A., Synak, P., Lewis, R. A., and Ras, Z. W. (2005). Extracting emotions from music data. In *ISMIS*, pages 456–465.

Zhou, Z.-H., Jiang, K., and Li, M. (2005). Multi-instance learning based web mining. *Appl. Intell.*, 22(2):135–147.

Zhou, Z.-H., Zhang, M.-L., Huang, S.-J., and Li, Y.-F. (2012). Multi-instance multi-label learning. *Artif. Intell.*, 176(1):2291–2320.