Rule Induction from a Decision Table Using Rough Sets Theory

Mert BAL
Mathematical Engineering Department
Yıldız Technical University
Istanbul, Turkey
mert.bal@gmail.com

Ayşe DEMİRHAN
Business Administration Department
Yıldız Technical University
Istanbul, Turkey
aysedemirhan74@gmail.com

Abstract— **Today, because of the rapid developments in both computer hardware and software industries, the increase in the storage capacities of huge databases, the data mining and the usage of the useful patterns that reside in the databases, became a very important research area. In parallel with the rapid increase in the data stored in the databases, effective use of the data is becoming a problem. To discover the rules or interesting and useful patterns from these stored data, the data mining techniques are used. If the data is incomplete or inaccurate, the results extracted from the database during the data discovery phase would be inconsistent and meaningless. Rough sets theory is a new mathematical approach used in the intelligent data analysis and data mining if data is uncertain or incomplete. This approach is of great importance in cognitive science and artificial intelligence, especially in machine learning, decision analysis, expert systems and inductive reasoning.**

There are many advantages of the rough set approach in intelligent data analysis. Some of these advantages are being suitable for parallel processing, finding minimal data sets, supplying effective algorithms to discover hidden patterns in data, valuation of the meaningfulness of the data, producing decision rule set from data, being easy to understand and the results obtained can be interpreted clearly. In the last years, the rough sets theory is widely used in different areas like engineering, banking and finance.

Keywords *- rough sets theory, data mining, decision table, rule discovery*

I. INTRODUCTION

Today, the size of the data stored in the databases of the organizations is growing each day and therefore we face difficulties about obtaining the valuable data. Databases are a collection of relational and non-recurring data to meet the demands of the organizations. Because the data stored in the databases is growing each day, it is getting harder for the users to reach the information. In the last few years, because of the rapid developments in both computer hardware and software industries, the increase in the storage capacities of huge databases, the data mining and the usage of the useful patterns that reside in the databases, became a very important research area. To discover the rules or interesting and useful patterns among these stored data in the databases, the Data Mining techniques are used. Storing huge amount of increasing data in the databases, which is called information explosion, it is necessary to transform these data into necessary and useful information. Using conventional statistics techniques fail to satisfy the requirements for analyzing the data, in the last years, the newly developed concepts Data Mining and Knowledge Discovery in Databases are getting more important. One of the approaches used in data mining and knowledge discovery is rough sets theory. According to this method, which is proposed in the beginning of 1980's, it is thought that knowledge can be obtained from every object in the universe.

In this study, a sample application about rule discovery from a decision table using rough sets theory is presented.

II. DATA MINING AND KNOWLEDGE DISCOVERY IN DATABASES

Data Mining is a discovery process of the hidden information from the data which is yet unknown and potentially useful. On the other hand, according to Raghavan and Sever, data mining discovers the general patterns and relations hidden in the data [1].

Decision rules are one of the widely used techniques to present the obtained information. A decision rule summarizes the relation between the properties. To transform the raw data residing in the database into valuable information, several stages of data processing is required. Data Mining is an iterative process that acts as a bridge the gap between logical decision-making and the data, and is possible the classification for finding the useful samples or using and combining the classification rules from the samples. This process combines the approaches used in different disciplines like machine learning, statistics, database management systems, data warehousing, and constraint programming [1].

In recent years many successful machine learning applications have been developed, in particular in domain of data mining and knowledge discovery. One of common tasks performed in knowledge discovery is classification. It consists of assigning a decision class label to a set of unclassified objects described

by a fixed set of attributes (features). Learning algorithms induce various forms of classification knowledge from learning examples, i.e., decision trees, rules, Bayesian classifiers. Decision rules are represented as logical expressions of the following form:
IF (conditions) THEN (decision class)
where conditions are formed as a conjunction of elementary tests on values of attributes. A number of various algorithms have already been developed to induce such rules.

Decision rules are one of the most popular type of knowledge used in practice; one of the main reasons for their wide application is their expressive and easily human-readable representation [2].

There are many successful applications of data mining process in many different areas. In Data mining applications, many methods to discover the useful patters are available and each method has advantages and disadvantages over the others. However, if needed, the advantages of different methods could be combined and hybrid methods could be created. The process of creating hybrid methods is a work of combining computational intelligence tools.

Many algorithms are used to implement a DM process. The reason is that some technologies result better than the others for different tasks, states and subjects do. In the core of the data mining lies a model creation process that represents a data set. A model creation process that represents a data set is generic for all DM products, on the other hand, the process itself is not generic.

Some methods used in DM processes are rough sets theory, Bayesian networks, genetic algorithms, neural networks, fuzzy sets and inductive logic programming.

DM functions are used to determine the pattern types that may exist in the DM tasks. Generally, DM tasks are classified into two categories: descriptive and estimator. Descriptive mining tasks characterize the general properties of the data in the database. On the other hand, estimator mining makes inferences from the available data to make estimations [3]. The samples of the DM functions and resulting discovered pattern types are classification, clustering, summarization, estimation, time series analysis, association rules, sequence analysis and visualization.

III. Rough Sets Theory

Rough sets theory is proposed by [4] in the beginning of 1980's and it is based on the assumption that a knowledge can be obtained from each object in the universe [5, 6].

In rough sets theory, the objects, characterized by the same information, have the same existing knowledge; this means they are indiscernible. Indiscernibility relationship produced using this way forms the mathematical basis of the rough sets theory. The sets of the same indiscernible objects are called "*elementary set*" and form the smallest building blocks (atoms) of the information about the universe. Some combinations of those elementary sets are called "*crisp set*", otherwise the set is called "*rough set*" Each rough set has boundary region. For example, like the unclassified objects with certainty. Significantly, rough sets, in contrast precise sets, cannot be characterized by the information of their elements. A rough set and a precise set pair are called the lower and upper approximation of the related rough set. Lower approximation contains all the objects belong to the set, but upper approximation contains the objects that may belong to the set. The differences between these lower and upper approximations define the boundary region of the rough set. The lower and the upper approaches are two basic functions in the rough sets theory.

There are many advantages of the rough sets approach in data analysis. Some of them are as follows.

- It finds minimal data sets and generates a decision rule from the resulting data.
- It performs the clear interpretations of the results and evaluation of the meaningfulness of the data.
- Many algorithms based on rough sets theory in particular are suitable for parallel processing [6].
- Rough sets can handle large volume of and any type of data. This capability is very useful in engineering analysis and modeling that several most appropriate solutions exist, that many sub-systems are managed by many variables and the relations exist between the sub-systems and affecting the performance of each other.
- Non-linear or discontinuous functional relations modeling capability supplies a strong method that can qualify the multi-dimensional and complex patterns. Because generated rules and used properties are not excessive, the patterns are concise, strong and sturdy. In addition, it supplies effective algorithms to find the hidden patterns in the data.
- Rough sets can identify and characterize the uncertain systems.
- Because the rough sets show the information as easy to understand logic patterns, where the inspection and validity of the data required or the decisions are taken by the rules and suitable for the supported situations, this method is successful [7].

IV. Basic Concepts of Rough Sets Theory

The basic concepts of rough sets theory are explained below.

A Information Systems

In rough sets theory, a data set is represented as a table and each row represents a state, an event or simply an object. Each column represents a measurable property for an object (a variable, an observation, etc.). This table is called an information system. More formally, the pair $\mathsf{A} = (U, \mathsf{A})$ represents an information system. U is a finite nonempty set that is called universe and A is a finite nonempty set of properties. Here, for $\forall a \in \mathsf{A}$, $a : U \rightarrow V_a$. The set V_a is called the value set of a. Another form of information systems is called decision systems. A decision system (i.e., decision table) expresses all the knowledge about the model.

A decision system is $\mathrm{A} = \left(U, \mathrm{A} \cup \{d\}\right)$ form of any information system. Here, $d \notin \mathrm{A}$ are decision attributes. Other attributes $a \in \mathrm{A} - \{d\}$ are called conditional attributes. Decision attributes can have many values, but usually they have a binary value like True or False [8, 9].

B Indiscernibility

Decision systems, which are a special form of information systems, contain all information about a model (event, state). In decision systems, the same or indiscernible objects might be represented more than once or the attributes may be too many. In this case, the resulting table will be bigger than desired. The relation about indiscernibility is as follows.

If a pair of relation $R \subseteq X \times X$ is either reflective (if an object relates to itself xRx), symmetrical (if xRy then yRx) or transitive (if xRy and yRz then xRz) then it is an equivalence relation. The equivalence class of $x \in X$ element contains all $y \in X$ objects, where xRy. Provided that $\mathrm{A} = (U, \mathrm{A})$ is an information system, then there is an equivalence relation between any $B \subseteq \mathrm{A}$ and a $IND_{\mathrm{A}}(B)$:

$$IND_{\mathrm{A}}(B) = \left\{(x, y) \in U^2 \mid \forall a \in B a(x) = a(y)\right\} \tag{1}$$

$IND_{\mathrm{A}}(B)$, $B-$ is called indiscernibility relation. If $(x, y) \in IND_{\mathrm{A}}(B)$, then the objects x and y are indiscernible with the attributes in B. The equivalence class of indiscernibility relation $B-$ is represented by $[x]_B$ [8, 10]. The indiscernibility relation $IND_{\mathrm{A}}(B)$ separates a universal set U, given as a pair of equivalence relation, into an $\{X_1, X_2, ..., X_r\}$ equivalence classes family. All equivalence classes family $\{X_1, X_2, ..., X_r\}$ defined by the relation $IND_{\mathrm{A}}(B)$ in set U forms a partition of set U and it is represented by B^*. The equivalence classes family B^* is called classification and represented by the expression $U / IND_{\mathrm{A}}(B)$. The objects belonging to the same equivalence classes X_i are indiscernible; otherwise, the objects are discernible by attributes subset B. The equivalence classes X_i, $(1,2,.........,r)$ of $IND_{\mathrm{A}}(B)$ relation are called elementary sets B in an information system A.

$[x]_B$ shows an elementary set B containing the element x and it is defined by the following equation (2):

$$[x]_B = \{y \in U \mid x IND_{\mathrm{A}} y\} \tag{2}$$

A sequenced pair $(U, IND_{\mathrm{A}}(B))$ is called approximation space. Any finite combination of elementary sets in an approximation space is called a set defined in the approximation space [7]. A elementary sets of an information system $\mathrm{A} = (U, \mathrm{A})$ are called the atoms of information system A.

C Discernibility Matrix

The study on the indiscernibility of the objects is carried out by [11]. In this study, indiscernibility function and indiscernibility matrix related to the creation of efficient algorithms for creating minimal feature subsystems sufficient to define all the aspects in a given information system are presented.

Let us assume that A is an information system that contains n number of objects. The indiscernibility matrix M_{A} for the information system A is a $n \times n$ symmetrical matrix, containing the elements c_{pq} shown below. Each element c_{pq} of this matrix comprises the attributes set that distinguishes the objects x_p and x_q.

$$c_{pq} = \left\{a \in \mathrm{A} \mid a(x_p) = a(x_q)\right\}, \quad (p, q = 1,2,..........., n) \tag{3}$$

Conceptually the indiscernibility matrix M_{A} is a $|U| \times |U|$ matrix. In order to generate the indiscernibility matrix, we should consider the different object pairs. Because $c_{pq} = c_{qp}$ and $c_{pp} = \emptyset$ for all objects x_p and x_q, it is not necessary to calculate half of the elements when generating the indiscernibility matrix M_{A}. That will lead to a reduction in computational complexity.

D Discernibility Function

Indiscernibility function is a function that defines how to distinguish an object or an object set from a certain subsystem of an object universe. Indiscernibility function is a multiplication of Boolean sums. The indiscernibility matrix M_{A} for any object $x \in U$, the indiscernibility matrix is generated as follows. Indiscernibility function f_{A} for an information system is a Boolean function of m number of Boolean variables $a_1^*, a_2^*, ..., a_m^*$ corresponding the attributes $a_1, a_2, ..., a_m$. Indiscernibility function f_{A} is expressed as follows:

$$f_{\mathrm{A}}\left(a_1^*, a_2^*, ..., a_m^*\right) = \wedge\left\{\vee c_{pq}^* \mid 1 \le q \le p \le n, c_{pq}^* \ne \emptyset\right\} \tag{4}$$

The formulas below are obtained:

$$c_{pq} = \left\{a \in \mathrm{A} \mid a(x_p) \ne a(x_q)\right\} \quad (p, q = 1,2,........., n) \tag{5}$$

$$c_{pq}^* = \left\{a^* \mid a \in c_{pq}\right\} \tag{6}$$

It might be possible to generate an indiscernibility function from an indiscernibility matrix M_A related to the object $x \in U$.

The function $f_A(x)$ is a multiplication function of the sum of Boolean variables $|A|$ while the variable a^* refers to the attribute a. Every combination of $f_A(x)$ comes from the object $y \in U$ that cannot be distinguished from x and each term in the combination represents the property that distinguishes one from another.

$$f_A(x) = \prod_{y \in U} \left\{ \sum a^* \mid a \in M_A(x, y) ve M_A(x, y) \neq \varnothing \right\} \quad (7)$$

The base contents of $f_A(x)$ in the universe U show the smallest subsets of A that is required distinguishing the objects from the object x.

V. Set Approximations

The basic idea underlying the rough sets theory is to generate the set approaches using the pair relation $IND_A(B)$. If X cannot be accurately defined using the attributes of A, then the lower and upper approximations are expressed. Let us assume that $A = (U, A)$ is an information system and $B \subseteq A$ and $X \subseteq U$. X can be approached only using the information contained in B, when X generates B-lower and B-upper approximations, represented by $\underline{B}X$ and $\overline{B}X$, respectively. Here, the lower and upper approximations are defines as follows:

$$\underline{B}X = \{x \mid [x]_B \subseteq X\} \quad (8)$$

$$\overline{B}X = \{x \mid [x]_B \cap X \neq \varnothing\} \quad (9)$$

The objects in $\underline{B}X$, B are classified certain members of X on the base of the information contained in B.The objects in $\overline{B}X$ can be classified probable members of X on the base of the information contained in B.

$$BN_B(X) = \overline{B}X - \underline{B}X \quad (10)$$

The equation (10) is called B-boundary region of X, and then it comprises the objects that cannot be classified certainly members of X on the base of the information contained in B. The set $U - \overline{B}X$ is called B-outside region of X, and it comprises the objects that certainly not belong to X on the base of the information contained in B.

If $BN_B(X) = \overline{B}X - \underline{B}X = \varnothing$, which is $\overline{B}X = \underline{B}X$, the set B is called certain set. $BN_B(X) \neq \overline{B}X - \underline{B}X \neq \varnothing$ If $\overline{B}X \neq \underline{B}X$, then the set B is called rough set. In this case, the set B can be qualified only with lower and upper approximations. Fig. 1 shows the lower and upper approximations of set X.

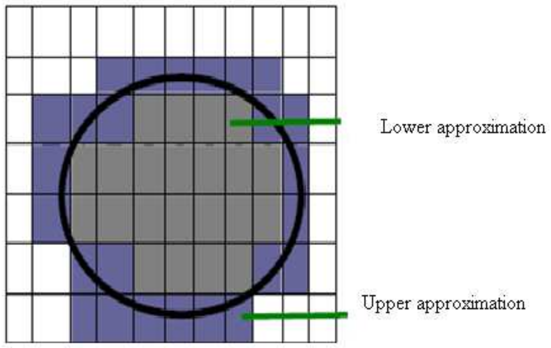

Figure 1. Upper and lower approximations of set X

The lower and upper approximations have the properties that are shown below:

1. $\underline{B}(X) \subseteq X \subseteq \overline{B}(X)$ (11)
2. $\underline{B}(\varnothing) = \overline{B}(\varnothing) = \varnothing$, $\underline{B}(U) = \overline{B}(U) = U$ (12)
3. $\overline{B}(X \cup Y) = \overline{B}(X) \cup \overline{B}(Y)$ (13)
4. $\underline{B}(X \cap Y) = \underline{B}(X) \cap \underline{B}(Y)$ (14)
5. $X \subseteq Y$ implies $\underline{B}(X) \subseteq \underline{B}(Y)$ and $\overline{B}(X) \subseteq \overline{B}(Y)$ (15)
6. $\underline{B}(X \cup Y) \supseteq \underline{B}(X) \cup \underline{B}(Y)$ (16)
7. $\overline{B}(X \cap Y) \subseteq \overline{B}(X) \cap \overline{B}(Y)$ (17)
8. $\underline{B}(-X) = -\overline{B}(X)$ (18)
9. $\overline{B}(-X) = -\underline{B}(X)$ (19)
10. $\underline{B}(\underline{B}(X)) = \overline{B}(\underline{B}(X)) = \underline{B}(X)$ (20)
11. $\overline{B}(\overline{B}(X)) = \underline{B}(\overline{B}(X)) = \overline{B}(X)$ (21)

Here, $-X$ means $U - X$.

One can define the following four basic classes of rough sets, i.e., four categories of vagueness:

- X is roughly B-definable, iff $\underline{B}(X) \neq \varnothing$ and $\overline{B}(X) \neq U$.
- X is internally B-undefinable, iff $\underline{B}(X) = \varnothing$ and $\overline{B}(X) \neq U$.
- X is externally B-undefinable, iff $\underline{B}(X) \neq \varnothing$ and $\overline{B}(X) = U$.
- X is totally B-undefinable, $\underline{B}(X) = \varnothing$ and $\overline{B}(X) = U$.

The intuitive meaning of this classification is the following.

X is roughly B-definable means that with the help of B we are able to decide for some elements of U that they belong to X and for some elements of U that they belong to - X.

X is internally B-undefinable means that using B we are able to decide for some elements of U that they belong to - X but we are unable to decide for any element U whether it belongs to X.

X is externally B-undefinable means that using B we are able to decide for some elements of U that they belong to X but we are unable to decide for any element U whether it belongs to - X.

X is totally B-undefinable means that using B we are unable to decide for some element of U whether it belongs to X or - X [8].

The universe can be divided into three disjoint regions using the upper and lower approximations, relating to any subset $X \subseteq U$. Boundary, positive and negative regions are described as below.

$$BND(X) = \overline{B}(X) - \underline{B}(X) \tag{22}$$

$$POS(X) = \underline{B}(X) \tag{23}$$

$$NEG(X) = U - \overline{B}(X) \tag{24}$$

A member of the negative region $NEG(X)$ does not belong to X. A member of the positive region $POS(X)$ belongs to X, and only one member of the boundary region $BND(X)$ belongs to X [12]. These regions are shown in the Fig. 2.

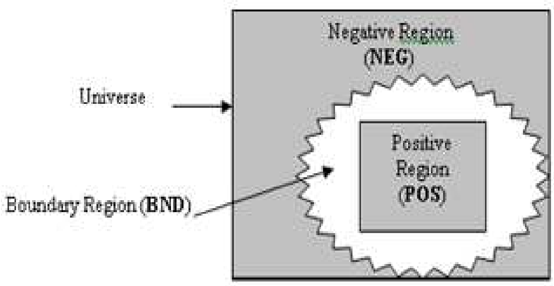

Figure 2. The negative, positive and the boundary regions of a rough set

A rough set can be characterized intuitively with the following coefficient:

$$\alpha_B(X) = \frac{|\underline{B}(X)|}{|\overline{B}(X)|} = \frac{card(\underline{B}(X))}{card(\overline{B}(X))} \tag{25}$$

Here the coefficient $\alpha_B(X)$ is called the accuracy of the approach and the number of members of the set $\overline{B}(X)$ is expressed as $|\overline{B}(X)|$ and the number of members of the set $\underline{B}(X)$ is expressed as $|\underline{B}(X)|$.

It is obvious that $0 \le \alpha_B(X) \le 1$. ($\alpha_B(X) \in [0,1]$)

If $\alpha_B(X) = 1$ then X is called certain relating to B, otherwise if $\alpha_B(X) < 1$ then X, is called rough relating to B.

VI. Rule Induction from Complete Decision Table

A decision table is an information system $T = (U, A \cup \{d\})$ such that each $a \in A$ is a condition attribute and $d \notin A$ is a decision attribute. Let V_d be the value set $\{d_1, d_2, ..., d_u\}$ of the decision attribute d. For each value $d_i \in V_d$, we obtain a decision class $U_i = \{x \in U \mid d(x) = d_i\}$ where $U = U_1 \cup U_2 \cup .... \cup U_{|V_d|}$ (i.e., $u = |V_d|$) and for every $x, y \in U_i$, $d(x) = d(y)$.

The B-positive region of d is defined by $POS_B(d) = \underline{B}(U_1) \cup \underline{B}(U_2) \cup .... \cup \underline{B}(U_{|V_d|})$.

A subset B of A is a relative reduct of T if $POS_B(d) = POS_A(d)$ and there is no subset B' of B with $POS_{B'}(d) = POS_A(d)$.

We define a formula $(a_1 = v_1) \wedge (a_2 = v_2) \wedge .... \wedge (a_n = v_n)$ in T (denoting the condition of a rule) where $a_j \in A$ and $v_j \in V_{a_j}$ $(1 \le j \le n)$. The semantics of the formula in T is defined by $\|(a_1 = v_1) \wedge (a_2 = v_2) \wedge .... \wedge (a_n = v_n)\|_T = \{x \in U \mid a_1(x) = v_1, a_2(x) = v_2, ..., a_n(x) = v_n\}$

Let φ be a formula $(a_1 = v_1) \wedge (a_2 = v_2) \wedge .... \wedge (a_n = v_n)$ in T.

A decision rule for T is of the form $\varphi \to (d = d_i)$, and it is true if $\|\varphi\|_T \subseteq \|d = d_i\|_T (= U_i)$.

The accuracy and coverage of a decision rule r of the form $\varphi \to (d = d_i)$ are respectively defined as follows:

$$\text{accuracy}\left(T,r,U_i\right)=\frac{\left|U_i \cap \left[\left|\varphi\right|\right]_T\right|}{\left|\left[\left|\varphi\right|\right]_T\right|} \tag{26}$$

$$\text{coverage }\left(T,r,U_i\right)=\frac{\left|U_i \cap \left[\left|\varphi\right|\right]_T\right|}{\left|U_i\right|} \tag{27}$$

In the evaluations $\left|U_i\right|$ is the number of objects in a decision class U_i and $\left|\left[\left|\varphi\right|\right]_T\right|$ is the number of objects in the universe $U = U_1 \cup U_2 \cup .... \cup U_{|V_d|}$ that satisfy condition φ of rule r. Therefore, $\left|U_i \cap \left[\left|\varphi\right|\right]_T\right|$ is the number of objects satisfying the condition φ restricted to a decision class U_i [13].

In this study, different kinds of rules are generated based on the characteristics from the decision table using ROSE2 (Rough Set Data Explorer) software.

ROSE2 is a modular software system implementing basic elements of the rough set theory and rule discovery techniques. It has been created at the laboratory of Intelligent Decision Support Systems of the Institute of Computing Science in Poznan.

ROSE2 software system contains several tools for rough set based knowledge discovery. These tools can be listed as below (http://idss.cs.put.poznan.pl/site/rose.html):

- data preprocessing, including discretization of numerical attributes,
- performing a standard and an extended rough set based analysis of data,
- search of a core and reducts of attributes permitting data reduction,
- inducing sets of decision rules from rough approximations of decision classes,
- evaluating sets of rules in classification experiments,
- using sets of decision rules as classifiers.

All computations are based on rough set fundamentals introduced by Pawlak [4]. To obtain the decision rules from the decision table, the algorithms LEM2 [14, 15, 16], Explore [17] and MODLEM [18] are utilized. LEM2, Explore and MODLEM algorithms for rule induction which are used in this study will be defined briefly as follows. These algorithms are strong for both complete and incomplete decision tables induction.

A LEM2 Algorithm

LERS [14] (LEarning from examples using Rough Set) is a rule induction algorithm that uses rough set theory to handle inconsistent data set, LERS computes the lower approximation and the upper approximation for each decision concept. LEM2 algorithm of LERS induces a set of certain rules from the lower approximation, and a set of possible rules from the upper approximation. The procedure for inducing the rules is the same in both cases [19]. This algorithm follows a classical greedy scheme which produces a local covering of each decision concept, i.e., it covers all examples from the given approximation using a minimal set of rules [20].

B MODLEM Algorithm

Preliminary discretization of numerical attributes is not required by MODLEM. The algorithm MODLEM handles these attributes during rule induction, when elementary conditions of a rule are created. MODLEM algorithm has two version called MODLEM-Entropy and MODLEM –Laplace. A similar idea of processing numerical data is also considered in other learning systems, i.e., C4.5 [21] performs discretization and tree induction at the same time. In general, MODLEM algorithm is analogous to LEM2. MODLEM also uses rough set theory to handle inconsistent examples and computes a single local covering for each approximation of the concept [19]. The search space for MODLEM is bigger than the search space for original LEM2, which generates rules from already discretized attributes. Consequently, rule sets induced by MODLEM are much simpler and stronger.

C Explore Algorithm

Explore is a procedure that extracts from data all decision rules that satisfy requirements, regarding i.e., strength, level of discrimination, length of rules, as well as conditions on the syntax of rules. It may also be adapted to handle inconsistent examples either by using rough set approach or by tuning a proper value of the discrimination level. Induction of rules is performed by exploring the rule space imposing restrictions reflecting these requirements. Exploration of the rule space is performed using a procedure which is repeated for each concept to be described. Each concept may represent a class of examples or one of its rough approximations in case of inconsistent examples. The main part of the algorithm is based on a breadth-first exploration which amounts to generating rules of increasing size, starting from one-condition rules. Exploration of a specific branch is stopped as soon as a rule satisfying the requirements is obtained or a stopping condition, reflecting the impossibility to fulfill the requirements, is met [20].

D Example

Let us assume that we have the following T complete decision table in Table I. In this table, U represents the universe, A represents the attributes, d represents the decision classes, and V represents the values that each attribute has.

$$U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}\}$$

$A = \{a_1, a_2, a_3, a_4\}$, $d = \{1,2,3\}$, $V_1 = \{1,2,3\}$, $V_2 = \{1,3\}, V_3 = \{1,2,3,4\}, V_4 = \{1,2,4,5\}$

TABLE I. A COMPLETE DECISION TABLE

U	a_1	a_2	a_3	a_4	d
x_1	1	3	2	5	1
x_2	2	1	3	4	2
x_3	1	3	2	5	1
x_4	1	1	2	2	1
x_5	3	3	4	1	2
x_6	3	3	1	2	3
x_7	2	1	3	4	2
x_8	3	3	4	1	2
x_9	1	3	2	5	1
x_{10}	1	3	2	5	2
x_{11}	3	3	1	2	3
x_{12}	2	1	3	4	3

Exact and approximate rules generated using algorithms LEM2, Explore and MODLEM (MODLEM-Entropy and MODLEM-Laplace) from the decision tables are shown below with IF-THEN.

rule 1: IF (a2 = 1) AND (a4 = 2) THEN (d = 1)
rule 2: IF (a3 = 4) THEN (d = 2)
rule 3: IF (a3 = 1) THEN (d = 3)
rule 4: IF (a4 = 5) THEN (d = 1) OR (d = 2)
rule 5: IF (a1 = 2) THEN (d = 2) OR (d = 3)
rule 6: IF (a3 = 4) THEN (d = 2)
rule 7: IF (a4 = 1) THEN (d = 2)
rule 8: IF (a3 = 1) THEN (d = 3)
rule 9: IF (a1 = 3) AND (a4 = 2) THEN (d = 3)
rule 10: IF (a2 = 3) AND (a4 = 2) THEN (d = 3)
rule 11: IF (a1 < 1.5) AND (a2 < 2) THEN (d = 1)
rule 12: IF (a3 >= 3.5) THEN (d = 2)
rule 13: IF (a3 < 1.5) THEN (d = 3)
rule 14: IF (a4 >= 4.5) THEN (d = 1) OR (d = 2)
rule 15: IF (a1 >= 1.5) AND (a2 < 2) THEN (d = 2) OR (d =3)
rule 16: IF (a1 < 1.5) AND (a2 < 2) THEN (d = 1)
rule 17: IF (a3 >= 3.5) THEN (d = 2)
rule 18: IF (a3 < 1.5) THEN (d = 3)
rule 19: IF (a4 >= 4.5) THEN (d = 1) OR (d = 2)
rule 20: IF (a1 >= 1.5) AND (a2 < 2) THEN (d = 2) OR (d =3)

Among these rules; Rule 1-Rule 5 are produced by LEM2, Rule 6-Rule 10 are produced by Explore algorithms, Rule 11-Rule 16 are produced by MODLEM-Entropy and finally Rule 16- Rule 20 are produced by MODLEM-Laplace algorithms.

VII. CONCLUSION

In parallel with the rapid developments in both computer hardware and software industries, the increase in the storage capacities of huge databases, the data mining and the usage of the useful patterns that are residing in the databases, became a very important research area. To discover the rules or interesting and useful patterns among these stored data, the data mining methods are used. Rules are one of the widely used techniques to present the obtained information. A rule defines the relation between the properties and gives a comprehensible interpretation. If the data is incomplete or inaccurate, the results extracted from the database during the data mining phase would be inconsistent and meaningless. Rough set theory is a new mathematical approach used in the intelligent data analysis and data mining if data is uncertain or incomplete.

In this study, the mathematical principles of the rough set theory are discussed and an application about rule discovery using rough set theory from a decision table is presented. LEM2, Explore and MODLEM algorithms in the software ROSE2 are used to discover these rules.

MODLEM algorithm has two version called MODLEM-Entropy and MODLEM –Laplace. In the given application, there are twelve elements in the universe. Considering that much more data exist in the real life problems, it can be seen that how important this method is to discover the interesting patterns.

Also, these algorithms have different approaches to the decision rules that are produced from decision tables and have strong characteristics comparing to each other.

REFERENCES

[1] Sever, H., Oğuz, B., “Veri Tabanlarında Bilgi Keşfine Formel Bir Yaklaşım: Kısım II- Eşleştirme Sorgularının Biçimsel Kavram Analizi ile Modellenmesi”, Bilgi Dünyası, 15-44, 2003. (In Turkish)

[2] Stefanowski, J., “Changing Representation of Learning Examples While Inducing Classifiers Based on Decision Rules”, Artificial Intelligence Methods, AI-METH 2003, Gliwice Poland, 2003.

[3] Han, J., Kamber, M., "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, San Francisco, 2001.

[4] Pawlak, Z., "Rough Sets", International Journal of Computer and Information Sciences 11, pp.341-356, 1982.

[5] Nguyen, H.S., Slezak, D., "Approximate Reducts and Association Rules Correspondence and Complexity Results", Proccedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining and Granular Computing, Vol.1711, LNCS, Springer Verlag, p.137-145, 1999.

[6] Pawlak, Z., "Rough Sets and Intelligent Data Analysis", Information Sciences, Vol.147, No.1, pp.1-12, 2002.

[7] Binay, H.S., "Yatırım Kararlarında Kaba Küme Yaklaşımı", Ankara Üniversitesi, Doktora Tezi, Ankara, 2002 (In Turkish).

[8] Komorowski, J., Polkowski, L., Skowron, A., "Rough Sets: A tutorial", Rough-Fuzzy Hybridization: A new Method for Decision Making, Spriger Verlag, 1998.

[9] Hui, S., "Rough Set Classification of Gene Expression Data", Bioinformatics Group Project, 2002.

[10] Komorowski, J., Pawlak, Z., Polkowski, L., Skowron, A., "A Rough Set Perspective on Data and Knowledge" 1999.

[11] Skowron, A. ve Rauszer, C., "The Discernibility Matrices and Functions in Information Systems", Intelligent Decision Support Handbook of Advances and Applications of the Rough Set Theory. Kluwer Academic Publishers, p.311–362, 1992.

[12] Allam, A.A., Bakeir, M.Y., Abo-Tabl, E.A., "New Approach for Basic Rough Set Concepts", International Conference on Rough Sets, Fuzzy Sets, Data Mining, Granular Computing, p.64-73, 2005.

[13] Kaneiwa K., "A Rough Set Approach to Mining Connections from Information System", In: Proceedings of the 25th ACM Symposium on Applied Computing, Switzerland, p.990-996, 2010.

[14] Grzymala-Busse, J.W., "LERS-A System for Learning from Examples Based on Rough Sets", In: Slowinski, R., Editor, Intelligent Decision Support Handbook of Application and Advances of the Rough Sets Theory, Kluwer Academic Publishers, p.3-18, 1992.

[15] Grzymala-Busse, J.W., Lakshmanan, A., "LEM2 with Interval Extension: An Induction Algorithm for Numerical Attributes" In: Tsumoto, S.,Editor, Proceeding of the fourth International Workshop on Rough Sets, Fuzzy Sets and Machine Discovery, Tokyo, p.67-73, 1996.

[16] Stefanowski,J., "On Rough Set Based Approaches to Induction of Decision Rules" In:Polkowski L., Skowron,A., Editors, Rough Sets in Data Mining and Knowledge Discovery,Vol.1,Physica Verlag, p.500-529, 1998.

[17] Mienko R., Stefanowski, J., Taumi, K., Vanderpooten, D., "Discovery-Oriented Induction of Decision Rules", Cahier du Lamsade, No.141, Université Paris Dauphine, 1996.

[18] Stefanowski, J., "The Rough Set Based Rule Induction Technique for Classification Problems", In: Proceedings of sixth European Conference on Intelligent Techniques and Soft Computing, EUFIT 98, Aachen, Germany, p.109-113, 7-10 September 1998.

[19] Grzymala-Busse, J.W., Stefanowski, J., Three Discretization Methods for Rule Induction", International Journal of Intelligent Systems, Vol.16, p.29-38, 2001.

[20] Stefanowski, J., Vanderpooten, D., "Induction of Decision Rules in Classification and Discovery-Oriented Perspectives", International Journal of Intelligent Systems, Vol.16, pp.13-27, 2001.

[21] Quinlan, J.R., "C4.5: Programs for Machine Learning", Morgan Kaufmann Publishers, San Francisco 1993.