Fundamenta Informaticae (2002) 323–344
IOS Press

On Modal and Fuzzy Decision Logics Based on Rough Set Theory *

Tuan-Fang Fan
Department of Information Engineering,
National Penghu Institute of Technology, Penghu, Taiwan.
Email: dffan@npit.edu.tw

Churn-Jung Liau [C]
Institute of Information Science,
Academia Sinica, Taipei, Taiwan.
Email: liaucj@iis.sinica.edu.tw

Yiyu Yao
Department of Computer Science,
University of Regina, Regina, Saskatchewan, Canada S4S 0A2.
Email: yyao@cs.uregina.ca

Abstract. Some modal decision logic languages are proposed for knowledge representation in data mining through the notions of models and satis£ability. The models are collections of data tables consisting of a £nite set of objects described by a £nite set of attributes. Some relationships may exist between data tables in a collection and the modalities of our languages are interpreted with respect to these relations in Kripkean style semantics. The notion of fuzzy decision logic is also reviewed and combined with the modal decision logic. The combined logic is shown to be useful in the representation of fuzzy sequential patterns.

Keywords: Data table, decision logic, modal decision logic, fuzzy decision logic, rough set theory

*This is a substantially revised version of [11]
[C]Corresponding author

1. Introduction

Recently, knowledge discovery in databases (KDD) and its kernel data mining have received more and more attention for practical applications. While the mainstream research of data mining concentrates on the design of ef£cient algorithms for extracting knowledge from databases, the question to close the semantic gap between structured data and human-comprehensible concepts has been a lasting challenge for the research community [25]. This is called the interpretability problem of intelligent data analysis in [25]. Since the discovered knowledge is useful for a human user only when he can understand its meaning, the knowledge representation formalism will play an important role in the utilization of the induced rules.

Many different forms of knowledge have been considered by the KDD researchers, notably, the association rules and sequential patterns [1, 2]. However, it is in general dif£cult to integrate the discovered patterns and traditional AI systems. The main reason is that the inference engine of AI systems usually employ a logic-based knowledge representation, which is quite different from the specialized patterns discovered by a £xed data mining algorithm. Therefore, a uniform interface between the discovery and utilization of knowledge is urgently needed. The interface will transform the discovered patterns into the knowledge based on the logical formalism employed by the AI system(Figure 1).

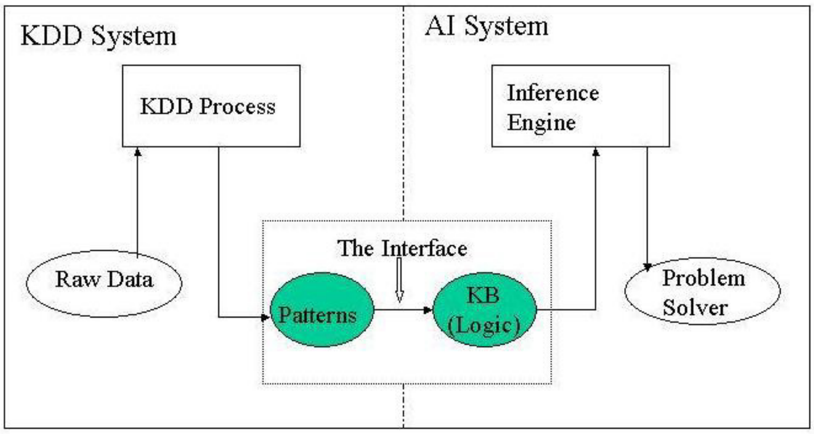

Figure 1. An interface is needed between the KDD and AI systems

The advantages of the logic-based representation for data mining have also been observed in the past [15].

> *. . . a coherent formalism, capable of dealing uniformly with induced knowledge and background, or domain, knowledge, would represent a breakthrough in the design and development of decision support systems, in diverse application domains. The advantages of such an integrated formalism are, in principle:*
>
> - *a high degree of expressiveness in specifying expert rules, or business rules;*
> - *the ability to formalize the overall KDD process, thus tailoring a methodology to a speci£c class of applications;*

- *the separation of concerns between the speci£cation level and the mapping to the underlying databases and data mining tools.*

The rough set theory proposed by Pawlak provides an effective tool for extracting knowledge from data tables [47]. In fact, many powerful data mining algorithms have been proposed based on the rough set theory(for example, see papers in [55, 56, 49] for some recent progress). To represent and reason about the extracted knowledge, a decision logic (DL) is also proposed in [47]. The semantics of the logic is de£ned in a Tarskian style through the notions of models and satisfaction.

Due to the following two reasons, DL is a good candidate to serve as the bridge between the KDD and AI systems: On the one hand, the data mining algorithms based on rough set theory usually extract rules which can be easily represented in the syntactical form of DL language. On the other hand, the semantic similarity between DL and Classical logic makes it easier to integrate the mined results into knowledge-based systems.

While DL can be considered as an instance of classical logic in the context of data tables, different generalizations of DL corresponding to some non-classical logics are also desirable from the knowledge representation viewpoint. For example, to deal with uncertain or incomplete information, some generalized decision logics have been proposed in [10, 31, 32, 63, 64].

These generalized decision logics, however, mostly focus on the representation of knowledge from a single data table. Though in principle, all data can be put into a single table, it is sometimes more natural to represent them by a collection of data tables. For example, in an enterprise database, the business transaction records may be stored as a collection of data tables indexed by dates. To extract knowledge from such structured data tables, we need richer representation languages than the decision logic. Among the traditional logical tools, modal logic would be one of the most appropriate candidates that can meet the requirement since it is a logic for reasoning about relations in a broad sense [4], whereas the knowledge extracted from multiple data tables is usually concerned with the relationship of objects across different tables. The objective of this paper is to present such a formulation of modal decision logics based on multiple data tables.

In the next section, we £rst review the decision logic proposed by Pawlak. A general modal decision logic(MDL) is presented in section 3, which is followed by three case studies. They are respectively the uncertain, epistemic, and temporal decision logic. In each case, the syntax and semantics of the logics are presented and some illustrative examples are given. In section 5, the notion of fuzzy decision logic is reviewed and combined with the modal decision logic. It is shown that the combined formalism provides a natural representation of fuzzy sequential patterns. Finally, the summary is given in the concluding section and some further research directions are also pointed out.

2. Review of Decision Logic

In data mining tasks, a data table(DT) is taken as a regular approach for the storage of data. A formal de£nition of data table is given in [47].

De£nition 1. A data table[1] is a triplet

$$T = (U, A, \{a_T \mid a \in A\})$$

[1]Also called knowledge representation system, information system, or attribute-value system

where

- U is a nonempty £nite set, called the universe,
- A is a nonempty £nite set of primitive attributes, and
- for each $a \in A$, $a_T : U \to V_a$ is a total function, where V_a is the domain of values for a. Usually, we will simply write a instead of a_T for the functions.

Given a data table T, we will denote its universe U and attribute set A by $Uni(T)$ and $Att(T)$ respectively.

In [47], a decision logic(DL) is proposed for the representation of the knowledge discovered from data tables. The logic is called decision logic because it is particularly useful in a special kind of data table, called *decision table*. A decision table is a data table $T = (U, C \cup D, \{a_T \mid a \in C \cup D\})$, where $Att(T)$ can be partitioned into two sets C and D, called condition attributes and decision attributes respectively. By data analysis, decision rules relating the condition and the decision attributes can be derived from the table. A rule is then represented as an implication between formulas of the logic. Nevertheless, for a general data table, the acronym DL can also denote *data logic*.

The basic alphabet of a DL consists of a £nite set of attribute symbols $\mathcal{A}$ and for $a \in \mathcal{A}$, a £nite set of value symbols $\mathcal{V}_a$. The syntax of DL is then de£ned as follows:

De£nition 2.

1. An atomic formula of DL is a descriptor (a, v), where $a \in \mathcal{A}$ and $v \in \mathcal{V}_a$.
2. The well-formed formulas (wff) of DL is the smallest set containing the atomic formulas and closed under the Boolean connectives $\neg$, $\wedge$, and $\vee$.

A data table $T = (U, A, \{a_T \mid a \in A\})$ is an interpretation for a given DL if there is a bijection $f : \mathcal{A} \to A$ such that for every $a \in \mathcal{A}$, $V_{f(a)} = \mathcal{V}_a$. Thus, by somewhat abusing the notation, we will usually denote an atomic formula as (a, v), where $a \in A$ and $v \in V_a$, if the data tables are clear from the context. Intuitively, each element in the universe of a data table corresponds to a data record and an atomic formula, which is in fact an attribute-value pair, describes the value of some attribute in a data record. Thus the atomic formulas (and so the wffs) can be veri£ed or falsi£ed in a data record. This gives rise to a satisfaction relation between the universe and the set of wffs.

De£nition 3. Given a DL and an interpretation $T = (U, A, \{a_T \mid a \in A\})$ for it, the satisfaction relation $\models$ between $x \in U$ and wffs of DL is de£ned inductively as follows:

1. $(T, x) \models (a, v)$ iff $a(x) = v$
2. $(T, x) \models \neg\varphi$ iff $(T, x) \not\models \varphi$
3. $(T, x) \models \varphi \wedge \psi$ iff $(T, x) \models \varphi$ and $(T, x) \models \psi$
4. $(T, x) \models \varphi \vee \psi$ iff $(T, x) \models \varphi$ or $(T, x) \models \psi$

If φ is a DL wff, the set $m_T(\varphi)$ de£ned by

$$m_T(\varphi) = \{x \in U \mid (T, x) \models \varphi\}, \tag{1}$$

is called the meaning of the formula φ in T. If T is understood, we simply write $m(\varphi)$.

A formula φ is said to be valid in a data table T, written $T \models \varphi$ or $\models \varphi$ for short when T is clear from the context, if and only if $m(\varphi) = U$. That is, ϕ is satis£ed by all individuals in the universe.

A DL wff talks about the properties of individuals in the universe, so it is satis£ed by some individuals but falsi£ed by the others. However, the mined knowledge is usually regarding the aggregated or statistical information of all individuals. Obviously, the wffs valid in a data table represent a kind of knowledge that can be induced from the table since they hold for all individuals. However, not all kinds of useful information are in the form of valid wffs. Sometimes, even probabilistic rules are very useful from the viewpoint of knowledge discovery. To quantify the usefulness of the mined rules, some measures have been proposed in [63, 65].

In contrast with DL, where extra meta-level measures must be attached to the wffs, these measures can also be internalized to the language by the so-called generalized quanti£ers [13, 26]. This is the approach adopted by the monadic observational predicate calculus(MOPC) in [16]. A wff in DL corresponds to the open formula of MOPC, however, there is no counterpart for the closed formulas of MOPC in DL as yet. To de£ne the corresponding extension in DL, let us call the above-de£ned DL wffs individual formulas and £x a set of unary and binary quanti£ers in advance, then the *aggregate formulas* for a data table T are de£ned by the following formation rules:

1. if φ is an individual formula and q is an unary quanti£er, then $(q)\varphi$ is an aggregate formula,

2. if φ and ψ are individual formulas and q is a binary quanti£er, then $(q)(\varphi, \psi)$ is an aggregate formula,

3. if φ and ψ are aggregate formulas, so are $\neg\varphi$, $\varphi \wedge \psi$, and $\varphi \vee \psi$

Sometimes, we will use the in£x notation $\varphi q \psi$ instead of $(q)(\varphi, \psi)$ for a binary quanti£er q. Each quanti£er q is interpreted by its truth function Tr_q according to [16]. For each unary quanti£er q, $Tr_q : N^2 \rightarrow \{0, 1\}$ is a 2-place function from natural numbers to $\{0, 1\}$ and for the binary one, $Tr_q : N^4 \rightarrow \{0, 1\}$ is a four-place function. Then the satisfaction of an aggregate formula with respect to a data table T is de£ned as follows:

1. $T \models (q)\varphi$ iff $Tr_q(|m(\varphi)|, |m(\neg\varphi)|) = 1$,

2. $T \models (q)(\varphi, \psi)$ iff $Tr_q(|m(\varphi \wedge \psi)|, |m(\varphi \wedge \neg\psi)|, |m(\neg\varphi \wedge \psi)|, |m(\neg\varphi \wedge \neg\psi)|) = 1$,

3. $T \models \neg\varphi$, $T \models \varphi \wedge \psi$, and $T \models \varphi \vee \psi$ are de£ned inductively as in the case of individual formulas.

Note that the classical quanti£ers $\forall$ and $\exists$ are de£ned with truth functions $Tr_\forall(n_1, n_2) = 1$ iff $n_2 = 0$ and $Tr_\exists(n_1, n_2) = 1$ iff $n_1 > 0$.

3. General Modal Decision Logic

Just like the models of DL are data tables, those for modal decision logic (MDL) will be structured sets of data tables.

De£nition 4. Let I and J be two £xed sets of indices, then a structured set of data tables (SSDT) is a pair

$$\mathcal{S} = (\{T_i \mid i \in I\}, \{R_j \mid j \in J\}),$$

where each T_i is a data table and each R_j is a binary relation over $\{T_i \mid i \in I\}$.

In this paper, we will consider only the SSDT $\mathcal{S} = (\{T_i \mid i \in I\}, \{R_j \mid j \in J\})$ satisfying the following assumptions:

- £xed attribute assumption:
$$\forall i, j \in I, Att(T_i) = Att(T_j),$$
namely, we assume the data tables in an SSDT are homogeneous.

- constant domain assumption:
$$\forall i, j \in I, Uni(T_i) = Uni(T_j).$$
In other words, we assume the set of individuals stays unchanged between different data tables.

- £nite table assumption: I is £nite. This is a practical assumption since we will consider only a £nite amount of data in the knowledge discovery process.

It seems that these assumptions are restrictive. However, the £rst two assumptions can be relaxed. We will discuss them further in the concluding section.

The syntax of MDL is an extension of DL with the following rule:

- if φ is an individual (resp. aggregate) formula, so are $[j]\varphi$ and $\langle j\rangle\varphi$ for any $j \in J$.

Given an SSDT $\mathcal{S} = (\{T_i \mid i \in I\}, \{R_j \mid j \in J\})$, the satisfaction of individual formulas are de£ned by

1. $(T_i, x) \models_{\mathcal{S}} [j]\varphi$ iff for all T such that $(T_i, T) \in R_j$, $(T, x) \models_{\mathcal{S}} \varphi$
2. $(T_i, x) \models_{\mathcal{S}} \langle j\rangle\varphi$ iff there exists T such that $(T_i, T) \in R_j$ and $(T, x) \models_{\mathcal{S}} \varphi$
3. the satisfaction of classical formulas is de£ned as in the case of DL.

The satisfaction of aggregate formulas can be analogously de£ned and is denoted by $T \models_{\mathcal{S}} \varphi$. An aggregate formula φ is said to be valid in an SSDT $\mathcal{S}$, denoted by $\models_{\mathcal{S}} \varphi$, if $T \models_{\mathcal{S}} \varphi$ for each data table in $\mathcal{S}$.

4. Case Studies

In MDL, there is a set of modal operators $[j]$ which are interpreted semantically by the binary relations R_j over the data tables of an SSDT, however, it remains unspeci£ed how the binary relations are constructed. In the following sections, we will study some cases in which the binary relations between data tables arise naturally from the application problems.

4.1. Uncertain decision logic

We consider the application of MDL to the problem regarding uncertain data tables. The approach we adopt here is somewhat related to that given in [45, 46].

De£nition 5. An uncertain data table is a triplet

$$T = (U, A, \{a_T \mid a \in A\})$$

where

- U and A are de£ned as in the standard data tables and
- for each $a \in A$, $a_T : U \to (2^V_a - \{\emptyset\})$ is a set-valued function, where V_a is the domain of values for a.

For each $x \in U$, $a_T(x)$ denotes the set of possible values for its attribute a. Since $a_T(x)$ may contain more than one values, this means that we do not have the exact knowledge about what the value is. In particular, if $a_T(x) = V_a$, then we have null information for the particular x on its attribute a. Given an uncertain data table $T = (U, A, \{a_T \mid a \in A\})$, a *possible realization* of T is a data table $T' = (U, A, \{a_{T'} \mid a \in A\})$ such that for any $x \in U$ and $a \in A$, $a_{T'}(x) \in a_T(x)$. Let $\Xi(T)$ denote the set of all possible realizations of T, then the SSDT for T is de£ned as

$$\mathcal{S} = (\Xi(T), R_u)$$

where R_u is the universal relation, i.e., for each T_i and $T_j \in \Xi(T)$, $(T_i, T_j) \in R_u$.

Thus the language of uncertain modal logic(UDL) contains only two modalities $[u]$ and $\langle u \rangle$ and we will denote them by the ordinary alethic modalities $\Box$ and $\Diamond$ respectively.

Example 1. The following table is simpli£ed from one example in [27] used in the evaluation of researchers for a leadership in a computer science grant. (We omit one attribute and replace the null value by the corresponding domain of values.)

Researcher	Talent	Grade	d
1	{math, cs}	{B, MSc, Ph.D}	good
2	{cs}	{Ph.D}	excel.
3	{math}	{MSc}	good
4	{math, phil.}	{B, MSc, Ph.D}	good

In this table, d denotes the decision attribute. There are in total 36 possible realizations for the uncertain data table. Among them is the following one:

Researcher	Talent	Grade	d
1	math	Ph.D	good
2	cs	Ph.D	excel.
3	math	MSc	good
4	math	Ph.D	good

Thus according to the semantics of UDL, the following aggregate formula can be veri£ed in each possible realization:

$$\Diamond\forall((\mathrm{Talent}, \mathrm{math}) \supset (\mathrm{d}, \mathrm{good})).$$

■

4.2. Epistemic decision logic

The epistemic decision logic arises naturally in the reasoning about data security in the KDD process. The main challenge is to protect personal sensitive information in the release of microdata set, i.e. a set of records containing information on individuals. To achieve this, the re-identi£cation of individuals must be avoided. In other words, it is necessary to prevent the possibility of deducing which record corresponds to a particular individual even if the explicit identi£er of the individual is not contained in the released information. This problem has previously been studied in depth [21, 22, 53, 54, 57].

Since useful knowledge can be induced from the data tables, it is desirable that they can be released to the public. To protect the privacy of the individuals whose personal information is contained in a data table, the attributes of the table can be divided into three sets. The £rst one consists of the *key attributes*, which can be used to identify whom a data record belongs to. Therefore, they are always masked off before the table is released. Since the key attributes uniquely determine the individuals, we can assume that they are associated with elements in the universe U and omit them hence forth. Second, we have a set of *public attributes*, the values of which are known to the public. For example, in [57], it is pointed out that some attributes like birth-date, gender, ethnicity, etc., are included in some public databases such as census data or voter registration lists. These attributes, if not appropriately generalized, may be used to re-identify an individual's record in a medical data table, and this will cause privacy leakage. The last kind of attribute is the *con£dential ones*, the values of which we have to protect. It is often the case that there is an asymmetry between the values of a con£dential attribute. For example, if the attribute is the HIV test result, then the revelation of a '$+$' value may cause serious privacy invasion, whereas it does not matter to know that an individual has a '$-$' value.

To formally state the data security problem, let $T_1 = (U_1, A, \{a_{T_1} \mid a \in A\})$ be a data table and $T_2 = (U_2, A \cup C, \{a_{T_2} \mid a \in A \cup C\})$ be an uncertain data table such that for each $c \in C$ and $x \in U_2$, $c(x)$ is a singleton. Then for any $B \subseteq A$, T_1 and T_2 are said to be B-linkable if there exists a bijection $\sigma : U_1 \to U_2$ such that $a_{T_1}(x) \in a_{T_2}(\sigma(x))$ for any $x \in U_1$ and $a \in B$. Note that if T_1 and T_2 are B-linkable, then they are also B'-linkable for any $B' \subseteq B$.

Given an A-linkable pair (T_1, T_2) and $B \subseteq A$, then each bijection σ mentioned above de£nes a B-linked con£guration which is a data table $T_\sigma = (U_1, A \cup C, \{a_{T_\sigma} \mid a \in A \cup C\})$ such that for each $x \in U_1$

1. if $a \in A$, then $a_{T_\sigma}(x) = a_{T_1}(x)$ and

2. if $c \in C$, then $c_{T_\sigma}(x) = c_{T_2}(\sigma(x))$.

In this formulation, table T_1 is assumed to be publicly available to everybody, so a user can know the A-values of each individual in U_1. On the other hand, table T_2 is the one to be (partially) released to the public. However, because the identity of each individual has been masked off, it is assumed that U_2 only contains some record serial numbers with which we can by no means identify the owner of the record. However, by linking the B-values between the two tables, it is possible to partially determine the owners of the records. A B-linked con£guration is such a linked mapping between U_1 and U_2 when only the sub-table of T_2 consisting of the columns $B \cup C$ is released.

Example 2. Let us consider the following two tables T_1 and T_2 for medical records, where $U_1 = \{a, b, c, d, e\}$, $U_2 = \{1, 2, 3, 4, 5\}$, $A = \{\text{Sex, Age}\}$, and $C = \{\text{HIV}\}$:

	Sex	Age
a	M	20
b	M	25
c	F	30
d	F	35
e	F	40

	Sex	Age	HIV
1	{M}	[20,30]	+
2	{M}	[20,30]	-
3	{F}	[25,35]	+
4	{F}	[30,40]	-
5	{F}	[30,40]	+

If $B = A$ or $B = \{\text{Sex}\}$, then all possible B-linked con£gurations are characterized by the HIV values of $\{a, b, c, d, e\}$ in the following table, so in total there are six B-linked con£gurations. Each column of the table corresponds to exactly a B-linked con£guration:

a	+	+	+	-	-	-
b	-	-	-	+	+	+
c	+	+	-	+	+	-
d	+	-	+	+	-	+
e	-	+	+	-	+	+

This means that if table T_2 or its sub-table consisting of the Sex and HIV columns only are released, then there are only six possible linked mappings between the individuals and the data records.

On the other hand, if $B = \{\text{Age}\}$, then in total there are eight $\{\text{Age}\}$-linked con£gurations characterized by the following table.

a	+	+	+	-	+	+	-	-
b	-	-	+	+	-	+	+	+
c	+	-	-	+	+	-	+	-
d	-	+	-	+	+	+	-	+
e	+	+	+	-	-	-	+	+

The effect of releasing only part of the uncertain data table is equivalent to making all values of the unreleased attributes null. In fact, the eight {Age}-linked con£gurations are the same as the {Sex, Age}-linked con£gurations for the two tables T_1 and T_2', where T_2' is as follows:

	Sex	Age	HIV
1	{M,F}	[20,30]	+
2	{M,F}	[20,30]	-
3	{M,F}	[25,35]	+
4	{M,F}	[30,40]	-
5	{M,F}	[30,40]	+

■

Given an A-linkable pair (T_1, T_2) as above, we can de£ne its epistemic SSDT as

$$\mathcal{S}(T_1, T_2) = (\mathcal{T}, \{R_B \mid B \subseteq A\})$$

where $\mathcal{T}$ is the set of all B-linked con£gurations for any $B \subseteq A$ and for each R_B and any two tables $T_i, T_j \in \mathcal{T}$, $(T_i, T_j) \in R_B$ iff T_i and T_j are both B-linked con£gurations. According to the semantics of general modal decision logic, $[B]\varphi$ means that φ can be known by data table linkage provided that the sub-table containing only the attributes $B \cup C$ is released. In general, we can stipulate some sensitive formulas which we would like to prevent the end user from knowing. Thus, if $[B]\varphi$ is true for some sensitive φ, then the release of such sub-table is unsafe.

Example 3. Continuing example 2, if the sensitive formula is $(HIV, +)$, then, since in any A-linked con£gurations T, $(T, x) \not\models_{\mathcal{S}} [A](HIV, +)$ for all $x \in U_1$, the release of the whole data table is safe. However, if we consider another sensitive formula $\varphi = (Sex, F) \Rightarrow_{0.5} (HIV+)$ where $\Rightarrow_{0.5}$ is a binary quanti£er de£ned by $Tr_{\Rightarrow_{0.5}}(n_1, n_2, n_3, n_4) = 1$ iff $\frac{n_1}{n_1+n_2} \geq 0.5$, then $T \models_{\mathcal{S}} [A]\varphi$ for any A-linked con£gurations T, so the release of the whole table is unsafe for the sensitive formula. To guarantee the safety, we can only release the sub-table consisting of the attributes Age and HIV. ■

4.3. Temporal decision logic

Perhaps the most useful instance of modal decision logics is the temporal one. There may be many variants of temporal decision logic. Here, we £rst formulate the most simple one based on linear time structure. The linear time structure can be mapped to an initial segment of the natural numbers and the main relations between time points are the "next" and "earlier-than" relations. Furthermore, we also need the universal relation for the formulation of sequential patterns in data mining.

De£nition 6. A (linear-time) temporal SSDT is of the form

$$\mathcal{S} = (\{T_i \mid 0 \leq i \leq n-1\}, \{R_+, R_<, R_u\})$$

where

- each T_i is a data table,

- $(T_i, T_j) \in R_+$ iff $j = i + 1$,
- $(T_i, T_j) \in R_<$ iff $i < j$, and
- $(T_i, T_j) \in R_u$ for all $0 \leq i, j \leq n - 1$.

The modalities $[+]$ and $[<]$ corresponds to the "next" and "future" operators in ordinary temporal logic and will be denoted by $\bigcirc$ and $\vec{\Box}$ respectively. The dual operator of $\vec{\Box}$(i.e. $\langle<\rangle$) is denoted by $\vec{\Diamond}$ as usual. Furthermore, we abbreviate a sequence of n modal operators $\bigcirc$ by $\bigcirc^n$. Also recall that $[u]$ and $\langle u\rangle$ are denoted by $\Box$ and $\Diamond$ as in uncertain decision logic.

The temporal decision logic may be applied to the mining of sequential patterns [2]. According to their de£nition, the sequential pattern mining problem is as follows:

> *Given a set of sequences, where each sequence consists of a list of elements and each element consists of a set of items, and given a user-speci£ed min support threshold, sequential pattern mining is to £nd all of the frequent subsequences, i.e., the subsequences whose occurrence frequency in the set of sequences is no less than min support.*

For example, in the analysis of customer purchase behavior, each sequence is the purchase history of a customer and each element of the sequence consists of all items purchased simultaneously by the customer at some time.

Example 4. To formulate the customer purchase behavior analysis, we can construct a temporal SSDT on the following way: The universe consists of all customers and the attributes are the items. Each attribute is bi-valued. Each data table contains the transaction records at some time. Thus if customer x purchased items b, c, e at time i, then, in table T_i, $b(x) = c(x) = e(x) = 1$ and $a(x) = 0$ for all other attributes a. A sequential pattern is in general represented as an individual formula in temporal decision logic:

$$\Diamond(\varphi_0 \wedge \vec{\Diamond}\, (\varphi_1 \wedge \vec{\Diamond}\, (\varphi_2 \cdots))) \tag{2}$$

where each φ_i is a conjunction of atomic formulas. To ensure the mining of frequent patterns, assuming the minimum support is $r \in [0, 1]$, we can use the aggregate formula $(r)\varphi$ where φ is an individual formula denoting a sequential pattern and r is a unary quanti£er de£ned by $Tr_r(m, n) = 1$ iff $\frac{m}{m+n} \geq r$. ■

Recently, sequential pattern mining was also used in the construction of intrusion detection rules [28, 29]. According to [29]:

> *The main techniques for intrusion detection are misuse detection and anomaly detection. For the former, the "signatures" of known attacks, i.e., the patterns of attack behavior and effects are used to identify a matched activity as an attack instance, whereas the latter uses established normal pro£les, i.e., the expected behavior, to identify any unacceptable deviation as possibly the result of an attack.*

In [29], the data mining technique is applied to a set of audit records. One kind of data they considered is the BSM data developed and distributed by MIT Lincoln Lab for the 1999 DARPA evaluation of intrusion detection systems. The data contains audit records of all *sendmail* sessions during a period of time. Each audit record corresponds to a UNIX system call made by *sendmail*. The attributes of each record include the system call name, the user and group IDs, the name of object accessed by the system call, arguments, etc. The expected patterns to be discovered is of the form $Pr(s_n \mid s_0, \cdots, s_{n-1})$ which is the probabilistic prediction of the $(n+1)$-th system call given the previous n system calls in a session.

Example 5. To model the intrusion detection application, we consider the universe U as the set of all sessions during a period of time. For the purpose of simpli£cation, we assume all sessions start at the same time. The attributes of each data table are just those for the system calls in the audit records. For $0 \leq i \leq n$, the data table T_i contains the system calls made at time i by each session. Then the expected patterns to be mined will be expressed by the following formula:

$$(\varphi_0 \wedge \bigcirc\varphi_1 \wedge \cdots \wedge \bigcirc^{n-1}\varphi_{n-1}) \Rightarrow_r \bigcirc^n \varphi_n \tag{3}$$

where each φ_i is an individual formula denoting the properties of system calls and $\Rightarrow_r$ is a binary quanti£er de£ned by $Tr_{\Rightarrow_r}(n_1, n_2, n_3, n_4) = 1$ iff $\frac{n_1}{n_1+n_2} \geq r$. ■

4.3.1. Dynamic decision logic

A variant of temporal decision logic is the dynamic decision logic. Sometimes, we may be interested in the effects of some actions. For example, the promotion of some items may be in¤uential to customers' purchase behavior with respect to the particular and other related items. Or, in the medical domain, some medical treatment may have certain effects on the test results of the patients. To formulate this kind of analysis, we need the dynamic decision logic.

Let Act be a set of actions, then the dynamic SSDT based on Act is

$$\mathcal{S}(Act) = (\{T_i \mid i \in I\}, \{R_\alpha \mid \alpha \in Act\})$$

where

- I is a £nite set of time points and T_i contains the data collecting at time i for $i \in I$
- for each $\alpha \in Act$ and $i, j \in I$, $(T_i, T_j) \in R_\alpha$ if α is carried out between time i and time j.

According to the semantics of MDL, a dynamic decision logic formula $[\alpha]\varphi$ means that φ necessarily holds after the action α is carried out.

Example 6. Let us consider again the customer purchase behavior analysis. If the action α denote "one-week sale of item a at a discount", then the induced patterns may be something like

$$(0.5)(a, 1) \supset [\alpha](0.8)(a, 1) \tag{4}$$

or

$$(b, 1) \Rightarrow_{0.6} [\alpha](a, 1). \tag{5}$$

Recall that (0.5) and (0.8) are unary quanti£ers and $\Rightarrow_{0.6}$ is a binary one. The formula at (4) means that if at least 50% of customers are purchasing item a, then after the promotion action, at least 80% of customers will be doing so. The formula at (5) means that if a and b are two related items, then the promotion of item a also has an effect on buyers of item b. In other words, 60% of customers purchasing b will be attracted by the price reduction of a (though they may still buy b at the same time). ■

5. Fuzzy Decision Logic and its Modal Extension

With the motivation of quantizing numerical attributes, a fuzzy decision logic is introduced in [10]. There are in general two kinds of attributes in a data table, the nominal ones and the numerical ones. The former is usually with £nite domains. For example, the status of a switch may be on or off, the sex of a person may be male or female, etc. On the other hand, numerical attributes often have an in£nite domain of values. Even though the domain is £nite, its cardinality may be very large. For example, the temperature may be a subset of real numbers.

Due to the continuity of the numerical domains, the objects which possess proximate values may behave similarly at their decision attributes. For example, two persons who have proximate ages may have a similar shopping behavior. Since data tables are £nite, not all possible values of the attributes appear in a table, so we should be able to extrapolate or interpolate the extracted rules to the values not appearing in the table.

To solve the data interpolation problem, many quantization techniques have been adopted [38]. The most direct one is the crisp quantization approach. By this technique, for an attribute a, we can partition V_a into n_a mutually disjoint subsets $D_1, D_2, \ldots, D_{n_a}$, and in the decision table, for each $x \in U$, $a(x)$ is replaced by D_j if $a(x) \in D_j$ for some $1 \leq j \leq n_a$. Although the quantization process may reduce the precision of the data, it also effectively hides irrelevant details of the data, so it is useful in summarizing the data. However, since the intervals do not necessarily correspond to natural language terms, the extracted rules lack a colloquial reading when we try to explain them. To obtain more meaningful quantization, we may in advance stipulate some linguistic terms as the labels of the resultant classes of the partition, and then the values in the domain are assigned to the respective classes according to the meaning of these linguistic terms. Thus semantics of natural language may guide the quantization process. However, even if some linguistic terms are given in advance, it is sometimes still dif£cult to decide the membership of some values. This is due to the fuzziness of these terms, so it is natural to interpret these terms as fuzzy sets instead of crisp ones. This means that the fuzzy quantization approach may be more appropriate for the problem. To represent the rules induced by the fuzzy quantization approach, we need a fuzzy decision logic (FDL).

The basic alphabet of FDL also consists of a £nite set of attribute symbols $\mathcal{A}$ and for $a \in \mathcal{A}$, a £nite set of linguistic terms $\mathcal{L}_a$ and the atomic formula of an FDL is now a descriptor (a, l_a) where $a \in \mathcal{A}$ and $l_a \in \mathcal{L}_a$. Then the formation rules of wffs for FDL are the same as those for DL. However, to interpret the wffs of an FDL in a data table, we have to £x a context for the linguistic terms.

It is well-known that many natural language terms are highly context-dependent. For example, the term "tall" may have quite different meanings for "a tall basketball player" and "a tall child". To model the context-dependency, we associate a context with each FDL. The context determines the domain of values of each attribute and assigns an appropriate meaning to each linguistic term. Formally, a context associated with an FDL is a pair $(\{V_a\}_{a \in \mathcal{A}}, ct)$, where V_a is a domain of values for each $a \in \mathcal{A}$ and ct is a function on the linguistic terms such that $ct(l_a) \in \tilde{\mathcal{P}}(V_a)$ if $l_a \in \mathcal{L}_a$, where $\tilde{\mathcal{P}}(V_a)$ denote the class of all fuzzy subsets of V_a. Henceforth, we assume a £xed context is given. By the £xed context, a data table $T = (U, A, \{a_T \mid a \in A\})$ is an interpretation for a given FDL if there is a bijection between $\mathcal{A}$ and A such that for every $a \in \mathcal{A}$, the linguistic terms in $\mathcal{L}_a$ are all mapped to fuzzy subsets of the domain for the corresponding attribute by the context.

Since each linguistic term is interpreted as a fuzzy subset of the attribute values, a data record may satisfy an individual formula in FDL to some degree. Thus the satisfaction between data records and individual formulas is no longer a qualitative relation.

De£nition 7. Let Φ denote the set of individual formulas of an FDL and a data table $T = (U, A, \{a_T \mid a \in A\})$ be an interpretation for the FDL, then the evaluation function $E_T : U \times \Phi \to [0, 1]$ is de£ned as follows:

1. $E_T(x, (a, l_a)) = \mu_{ct(l_a)}(a(x))$, where $\mu_{ct(l_a)}$ is the membership function of the fuzzy set $ct(l_a)$
2. $E_T(x, \neg\varphi) = 1 - E_T(x, \varphi)$
3. $E_T(x, \varphi \wedge \psi) = E_T(x, \varphi) \otimes E_T(x, \psi)$
4. $E_T(x, \varphi \vee \psi) = E_T(x, \varphi) \oplus E_T(x, \psi)$

where $\otimes : [0, 1] \times [0, 1] \to [0, 1]$ is a t-norm[2] and $\oplus$ is the t-conorm de£ned by $a \oplus b = 1-(1-a)\otimes(1-b)$

The meaning function of FDL can be de£ned as a mapping of an individual formula to a fuzzy subset of the universe. Let m_T denote the meaning function for T and φ is an individual formula, then

$$\mu_{m_T(\varphi)}(x) = E_T(x, \varphi) \tag{6}$$

for all $x \in Uni(T)$. The cardinality of a fuzzy subset X of the universe U is de£ned by the so-called Σ-count [23], i.e.

$$|X| = \sum_{x \in U} \mu_X(x). \tag{7}$$

The semantics of aggregate formulas can now be de£ned analogously as in the case of DL with the help of fuzzy cardinality. However, the truth functions for unary and binary quanti£ers are now respectively $Tr_q : \Re^2 \to \{0, 1\}$ and $Tr_q : \Re^4 \to \{0, 1\}$. Note that according to the semantics, the aggregate formulas are still two-valued, whereas the individual formulas are many-valued, so for an aggregate formula φ of FDL and a data table T, we can still write $T \models \varphi$ for its satisfaction. Other possibilities for de£ning the semantics of FDL aggregate formulas exist in [16], however, we only need the above de£nition for the purpose of this paper.

5.1. Fuzzy modal decision logic and its applications

What differentiates FDL and DL is their semantics. The syntax of DL and FDL is the same, so is the syntax of MDL and fuzzy modal decision logic (FMDL). The evaluation function for individual formulas of FDL is extended to the modal case as follows: Let $\mathcal{S} = (\{T_i \mid i \in I\}, \{R_j \mid j \in J\})$ be an SSDT, then the evaluation function $E_\mathcal{S} : I \times U \times \Phi \to [0, 1]$ is de£ned by

1. $E_\mathcal{S}(i, x, \varphi) = E_{T_i}(x, \varphi)$ if φ is an FDL individual formula and E_{T_i} is the evaluation function as de£ned above.
2. $E_\mathcal{S}(i, x, [j]\varphi) = \bigotimes\{E_\mathcal{S}(k, x, \varphi) \mid k \in I, (T_i, T_k) \in R_j\}$
3. $E_\mathcal{S}(i, x, \langle j\rangle\varphi) = \bigoplus\{E_\mathcal{S}(k, x, \varphi) \mid k \in I, (T_i, T_k) \in R_j\}$

[2] A binary operation $\otimes$ is a t-norm iff it is associative, commutative, and increasing in both places, and $1 \otimes a = a$ and $0 \otimes a = 0$ for all $a \in [0, 1]$.

where $\otimes$ and $\oplus$ are respectively the t-norm and t-conorm mentioned in de£nition 7. We use the t-norm and t-conorm in the semantics of modal formulas because $[j]$ and $\langle j\rangle$ are respectively considered as conjunctive and disjunctive on the set of tables. Analogously, we can also de£ne the meaning function m_{T_i} as a mapping from the individual formulas of FMDL to fuzzy subsets of $Uni(T_i)$ for each $i \in I$. Then the satisfaction of aggregate formulas can be de£ned by using the fuzzy cardinality for the non-modal cases and the semantic de£nition of MDL for the modal cases.

A direct application of the FMDL is the representation of fuzzy sequential patterns. In [18, 20], fuzzy data mining algorithms are proposed to deal with the discovery of fuzzy association rules from quantitative data. While the conventional association rule mining algorithms identify the simultaneous occurrence of some events, the fuzzy association rule mining algorithms are also concerned with how many times the events occur. The last-mentioned algorithms are further extended to £nding fuzzy sequential patterns from multiple-items transactions in [19]. In this subsection, we show that fuzzy sequential patterns can be easily represented as FMDL wffs. To facilitate such representation, we need only consider the temporal case. In fact, a fuzzy sequential pattern is syntactically the same as the one shown in (2). What is different is the computation of its support from the SSDT.

Example 7. Let us consider the following sequences of transaction data tables $\mathcal{S}$,

T_1	a	b	c	d
1	3	8	0	0
2	0	5	8	0
3	0	0	0	0
4	0	0	0	0
5	0	7	3	3

T_2	a	b	c	d
1	0	2	6	0
2	0	0	5	5
3	4	0	9	0
4	1	8	3	0
5	0	4	4	0

T_3	a	b	c	d
1	0	0	0	9
2	0	0	0	0
3	0	0	2	10
4	5	0	6	0
5	0	0	0	0

T_4	a	b	c	d
1	0	0	0	0
2	0	0	0	0
3	0	12	0	0
4	0	0	2	7
5	0	0	5	3

and from [20], we borrow the membership function for three linguistic terms "Low", "Middle", and "High" as in £gure 2. Now, the representation of the following fuzzy sequential pattern: "purchasing high volume of item b followed by purchase of middle amount of c", in the FMDL is:

$$\varphi = \Diamond((b, High) \wedge \overrightarrow{\Diamond}\, (c, Middle)).$$

Assume that the minimum support is 0.3, so we would like to know whether the aggregate formula $(0.3)\varphi$ holds in some T_i. In this example, we assume the t-norm and t-conorm are min and max respectively. Then, according to the semantics, for $1 \leq i \leq 4$,

$$E_{\mathcal{S}}(i, 1, \varphi) = \min(\mu_{High}(8), \mu_{Middle}(6)) = \min(0.4, 1) = 0.4$$

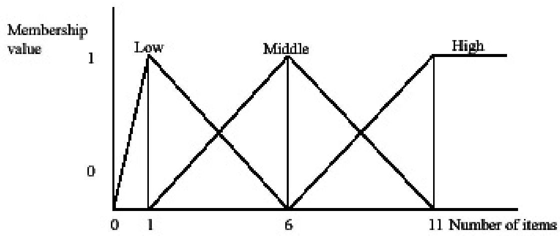

Figure 2. The membership functions for the linguistic terms

$$E_{\mathcal{S}}(i,2,\varphi) = \min(\mu_{High}(5), \mu_{Middle}(5)) = \min(0, 0.8) = 0$$

$$E_{\mathcal{S}}(i,3,\varphi) = 0$$

$$E_{\mathcal{S}}(i,4,\varphi) = \min(\mu_{High}(8), \max(\mu_{Middle}(6), \mu_{Middle}(2))) = \min(0.4, \max(1, 0.2)) = 0.4$$

$$\begin{aligned} E_{\mathcal{S}}(i,5,\varphi) &= \max(\min(\mu_{High}(7), \max(\mu_{Middle}(4), \mu_{Middle}(5))), \min(\mu_{High}(4), \mu_{Middle}(5))) \\ &= \max(\min(0.2, \max(0.6, 0.8)), \min(0, 0.8)) = 0.2 \end{aligned}$$

Thus the fuzzy cardinality of the meaning function $|m_{T_i}(\varphi)| = 0.4 + 0.4 + 0.2 = 1$ and $|m_{T_i}(\neg\varphi)| = 5 - 1 = 4$. Consequently, the aggregate formula $(0.3)\varphi$ does not hold at any T_i, i.e.,

$$T_i \not\models_{\mathcal{S}} (0.3)\varphi,$$

since $Tr_{0.3}(1,4) = 0$ due to $\frac{1}{1+4} < 0.3$. ■

6. Conclusion

Just like DL is used in the knowledge representation for data mining of a single data table, the MDL provides a uniform framework for representing knowledge mined from a collection of multiple data tables. The sets of data tables are structured in the sense that some relationship exists between their elements. We interpret the MDL formulas in such structured sets of data tables. In particular, the modalities are interpreted with respect to the relations between the data tables according to the Kripke semantics. Three instances of MDL are presented to illustrate the application potentials of the MDL representation formalism. Th combination of MDL and FDL is also proposed and its use in the representation of fuzzy sequential patterns is shown by an example.

6.1. Related works

Orłowska has been one of the £rst logicians proposing the modal logic systems for Pawlak's information systems [40, 39, 42, 41, 43, 44, 37]. Many excellent works on the modal logic systems for rough set theory and data tables have also been done by Demri, Düntsch, Rasiowa and Skowron, Vakarelov, and others [3, 5, 6, 7, 8, 9, 24, 51, 50, 52, 58, 60, 59, 61][3]. Some of these works, in particular those of Demri,

[3]The list is by no means exhaustive. For further references, see for examples [30, 48].

have explored the computational properties of the logical systems, while some of them provide complete axiomatization of such logics. Düntsch has also developed some systems from both the algebraic and logical aspects. All these works show the close connection between modal logic and Pawlak's information systems. This paper can be seen as a followup work of these previous works. However, while these works mainly deal with the logic for a single data table, we concentrate on the semantic structure of multiple data tables. Therefore, the accessibility relations in the Kripke semantics for the previous systems are usually de£ned between the individuals of a single data table, whereas in our semantic structures, the accessibility relations are de£ned between different data tables.

There is some similarity between our notion of SSDT and the relational information systems(RIS) proposed in [62]. An RIS is a triple $(\mathcal{A}, \mathcal{R}, A_0)$, where $\mathcal{A}$ is a family of data tables (maybe with different attributes and domains), $\mathcal{R}$ is a set of relations between the domains of data tables in $\mathcal{A}$, and A_0 is a distinguished data table in $\mathcal{A}$. Both SSDT and RIS deal with the structures of multiple data tables. It can be seen that the basic difference between SSDT and RIS is that the former imposes the relations between data tables, whereas the latter has its relations between the objects. In this sense, the binary relations for RIS are £ner than those for SSDT. A general adaptive scheme is also proposed for the mining of rules from RIS. However, no logical formalisms are developed for such structures. Therefore, the works in [62] should be complementary with ours.

It is also interesting to note the relationship between our work and the granular computing(GrC) model proposed in [33, 34, 35, 36]. In the GrC model, a binary relation between two universe is considered as essential. From the viewpoint of SSDT, this is a binary relation between two data tables. However, again, the relations of GrC models are imposed between objects instead of tables and no logical formalisms have been proposed for such models.

6.2. Future works

For simpli£cation of semantics, we have imposed some restrictive assumptions for the SSDT in the development of MDL. Therefore, further investigation is needed to lift the restriction.

For the £xed attribute assumption, if we allow the unde£ned value $\perp$ for every attribute, the assumption will not cause any loss of generality since we can assume that all data tables virtually have the same set of attributes. If an attribute really does not exist in one data table, the values of this speci£c attribute for individuals in that data table are all $\perp$. In this way, we can force all data tables to have the same set of attributes though some of them may be only virtually existing on some data tables. Nevertheless, to interpret the formulas in our logic, we must take the $\perp$ value into account and this will somewhat complicate the semantics of the logic, so we adopt the assumption for the purpose of simpli£cation. An analogous approach has been proposed in [12] for studying the modal logic formulation of databases.

As for the constant domain assumption, this means that we do not allow the birth and death of data records in different data tables. The same assumption has been made for some systems of modal predicate logic [14], so the well-known Barcan formula

$$\forall [j]\varphi \supset [j]\forall\varphi$$

holds in MDL for any modality $[j]$. This assumption can be replaced by the more relaxed increasing domain assumption. That is,

$$\forall i, k \in I, j \in J \text{ if } (T_i, T_k) \in R_j \text{ then } Uni(T_i) \subseteq Uni(T_k).$$

This means that the birth of new records is allowed though the records will never disappear along the direction of any binary relation. This assumption does not cause loss of generality any more since, if one record disappears in some data table, we can replace its attribute values by the $\perp$ value.

Another important research problem is the development of data mining algorithms based on the proposed logics. Since we mainly concentrate on the representation formalisms in this paper, the algorithmic or computational aspects of data mining have been largely ignored. However, it is indeed possible to develop some data mining algorithms with results representable in MDL or FMDL. The adaptive classi£cation algorithm proposed in [62] provides a practical direction along which the data mining tasks on multiple data tables can be done.

Also, we are currently working on the data mining algorithms for temporal decision logic. The basic idea is to employ the rough set-based algorithms or GUHA methods[17] to discover the rules in each single data table. The mined rules are represented by DL aggregate formulas. Let $\mathcal{MT}(T)$ denote the set of mined rules from the single data table T. For a data table T, we can compute the set

$$T^{\rightarrow} = \bigcap_{i \in I, (T, T_i) \in R_<} \mathcal{MT}(T_i).$$

By the mined rules, we mean that $\varphi \in \mathcal{MT}(T)$ implies $T \models_{\mathcal{S}} \varphi$. Therefore, if $\psi \in T^{\rightarrow}$, then $T \models_{\mathcal{S}} \vec{\Box}\, \psi$. Now if φ is an aggregate formula, then the set

$$\bigcap_{T \models_{\mathcal{S}} \varphi} T^{\rightarrow}$$

contains all mined rules ψ such that $\varphi \supset \vec{\Box}\, \psi$ is a mined rule for the whole SSDT.

This is only one of many possible forms of rules which can be discovered from multiple data tables, so it remains to be seen what kinds of rules are interesting from a KDD perspective.

Acknowledgement

The authors would like to express their deep thanks to Prof. Tsau-Young Lin (T.Y. Lin) and the anonymous referee for valuable comments by which this paper and its presentation are improved substantially.

References

[1] R. Agrawal and R. Srikant. “Fast algorithms for mining association rules”. In *Proc. 1994 Int. Conf. Very Large Data Bases (VLDB94)*, pages 487–499, 1995.

[2] R. Agrawal and R. Srikant. “Mining sequential patterns”. In *Proc. 1995 Int. Conf. Data Engineering (ICDE95)*, pages 3–14, 1995.

[3] Ph. Balbiani. “Axiomatization of logics based on Kripke models with relative accessibility relations”. In E. Orłowska, editor, *Incomplete Information: Rough Set Analysis*, pages 553–578. Physica-Verlag, 1997.

[4] P. Blackburn. “Representation, reasoning, and relational structures: a hybrid logic manifesto”. *Logic Journal of IGPL*, 8(3):339–365, 2000.

[5] S. Demri. “A class of information logics with a decidable validity problem”. In W. Penczek and A. Szalas, editors, *Proceedings of the 21st Symposium on Mathematical Foundations of Computer Science*, LNCS 1113, pages 291–302. Springer-Verlag, 1996.

[6] S. Demri. “A class of decidable information logics”. *Theoretical Computer Science*, 195(1):33–60, 1998.

[7] S. Demri and E. Orłowska. “ Informational representability of models for information logics”. In E.Orłowska, editor, *Logic at Work*, pages 383–409. Physica-Verlag, 1999.

[8] I. Düntsch. “A logic for rough sets”. *Theoretical Computer Science*, 179(1-2):427–436, 1997.

[9] I. Düntsch and G. Gediga. “Logical and algebraic techniques for rough set data analysis”. In L. Polkowski, S. Tsumoto, and T.Y. Lin, editors, *Rough Set Methods and Applications*, pages 521–544. Physica-Verlag, 2000.

[10] T.F. Fan, W.C. Hu, and C.J. Liau. “Decision logics for knowledge representation in data mining”. In *Proceedings of the 25th Annual International Computer Software and Applications Conference(COMPSAC)*, pages 626–631. IEEE Press, 2001.

[11] T.F. Fan, C.J. Liau, and Y.Y. Yao. “On modal decision logics”. In T.Y. Lin and C.J. Liau, editors, *Proceedings of the PAKDD’02 Workshop on the Foundation of Data Mining*, pages 21–26, 2002.

[12] M. Fitting. “Modality and databases”. In R. Dyckhoff, editor, *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*, LNCS 1847, pages 19–39. Springer-Verlag, 2000.

[13] P. Gärdenfors, editor. *Generalized Quanti£ers*. Dordrecht: Reidel Publishing Company, 1987.

[14] J.W. Garson. “Quanti£cation in modal logic”. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic, Vol II: Extensions of Classical Logic*, pages 249–307. D. Reidel Publishing Company, 1984.

[15] F. Giannotti, G. Manco, D. Pedreschi, and F. Turini. “Experiences with a logic-based knowledge discovery support environment”. In E. Lamma and P. Mello, editors, *AI*IA 99: Advances in Arti£cial Intelligence, 6th Congress of the Italian Association for Arti£cial Intelligence*, LNAI 1792, pages 202–213. Springer-Verlag, 1999.

[16] P. Hájek. “Logics for data mining (GUHA rediviva)”. *Neural Network World*, 10:301–311, 2000.

[17] P. Hájek and T. Havránek. *Mechanising Hypothesis Formation–Mathematical Foundations for a General Theory*. Springer-Verlag, 1978.

[18] T.P. Hong, K.Y. Lin, and S.L. Wang. “Mining generalized association rules from quantitative data”. In *Proceedings of The International Workshop on Intelligent Systems Resolutions in The Eighth Bellman Continuum*, pages 75–78, 2000.

[19] T.P. Hong, K.Y. Lin, and S.L. Wang. “Mining fuzzy sequential patterns from multiple-items transactions”. In *Proceedings of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, pages 1317–1321, 2001.

[20] T.P. Hong, K.Y. Lin, and S.L. Wang. “Mining a complete set of interesting generalized fuzzy association rules”. In T.Y. Lin and C.J. Liau, editors, *Proceedings of the PAKDD’02 Workshop on the Foundation of Data Mining*, pages 27–32, 2002.

[21] T.-s. Hsu, C.-J. Liau, and D.-W. Wang. A logical model for privacy protection. In *Proceedings of the 4th International Conference on Information Security*, LNCS 2200, pages 110–124. Springer-Verlag, 2001.

[22] A.J. Hundepool and L.C.R.J. Willenborg. “μ- and τ-ARGUS: Software for statistical disclosure control”. In *Proceedings of the 3rd International Seminar on Statistical Con£dentiality*, 1996.

[23] A. Kandel. *Fuzzy Mathematical Techniques with Applications*. Addison-Wesley Publishing Co., 1986.

[24] B. Konikowska. "A formal language for reasoning about indiscernibility ". *Bulletin of the Polish Academy of Science: Mathematics*, 35:239–249, 1987.

[25] R. Kruse, C. Borgelt, and D. Nauck. "Fuzzy data analysis: challenges and perspectives". In *Proceedings of the 8th IEEE International Conference on Fuzzy Systems*, pages 1211–1216, San Francisco, CA, 1999. IEEE.

[26] M. Krynicki, M. Mostowski, and L.W. Szczerba, editors. *Quanti£ers: Logics, Models and Computation*. Kluwer Academic Publishers, 1995.

[27] M. Kryszkiewicz and H. Rybiński. "Reducing information systems with uncertain attributes". In Z. W. Raś and M. Michalewicz, editors, *Proceedings of the 9th International Symposium on Methodologies for Intelligent Systems*, LNAI 1079, pages 285–294. Springer-Verlag, 1996.

[28] W. Lee, S.J. Stolfo, and K.W. Mok. A data mining framework for building intrusion detection models. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, pages 120–132, 1999.

[29] W. Lee and D. Xiang. Information-theoretic measures for anomaly detection. In *Proceedings of the 2001 IEEE Symposium on Security and Privacy*, pages 130–143, 2001.

[30] C.J. Liau. "An overview of rough set semantics for modal and quanti£er logics". *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, 8(1):93–118, 2000.

[31] C.J. Liau and D.R. Liu. "A logical approach to fuzzy data analysis". In J.M. Zytkow and J. Rauch, editors, *Proceedings of the Third European Conference on Principles of Data Mining and Knowledge Discovery*, LNAI 1704, pages 412–417. Springer-Verlag, 1999.

[32] C.J. Liau and D.R. Liu. "A Possibilistic decision logic with applications". *Fundamenta Informaticae*, 46(3):199–217, 2001.

[33] T.Y. Lin. Neighborhood systems and relational database(abstract). In *Proceedings of the 6th Annual computer Science Conference*, page 725, 1988.

[34] T.Y. Lin. Neighborhood systems and approximation in database and knowledge base systems. In *Proceedings of the 4th International Symposium on Methodologies of Intelligent Systems(Poster Session)*, pages 75–86, 1989.

[35] T.Y. Lin. "Granular computing on binary relations I: Data mining and neighborhood systems". In A. Skoworn and L. Polkowski, editors, *Rough Sets In Knowledge Discovery Vol 1: Methodology and Applications*, pages 107–121. Physica-Verlag, 1998.

[36] T.Y. Lin. "Data mining and machine oriented modeling: A granular computing approach". *Journal of Applied Intelligence*, 13(2):113–124, 2000.

[37] L. Fari nas del Cerro and E. Orłowska. "DAL–A logic for data analysis". *Theoretical Computer Science*, 36:251–264, 1985.

[38] H.S. Nguyen and S.H. Nguyen. "Discretization methods in data mining". In L. Polkowski and A. Skowron, editors, *Rough Sets in Knowledge Discovery*, pages 451–482. Physica-Verlag, 1998.

[39] E. Orłowska. "Logic of indiscernibility relation". *Bulletin of the Polish Academy of Science: Mathematics*, 33:473–485, 1985.

[40] E. Orłowska. "Logic of nondeterministic information". *Studia Logica*, XLIV:93–102, 1985.

[41] E. Orłowska. "Reasoning about vague concepts". *Bulletin of the Polish Academy of Science: Mathematics*, 35:643–652, 1987.

[42] E. Orłowska. "Semantics of knowledge operators". *Bulletin of the Polish Academy of Science: Mathematics*, 35:255–264, 1987.

[43] E. Orłowska. "Logic for reasoning about knowledge". *Zeitschrift f. Math. Logik und Grundlagen der Math*, 35:559–572, 1989.

[44] E. Orłowska. "Kripke semantics for knowledge representation logics". *Studia Logica*, XLIX:255–272, 1990.

[45] Z. Pawlak. "Information systems—theoretical foundations". *Information Systems*, 6(3):205–218, 1981.

[46] Z. Pawlak, editor. *Information Systems – Theoretical Foundations(in Polish)*. WNT, 1983.

[47] Z. Pawlak. *Rough Sets–Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, 1991.

[48] Z. Pawlak, L. Polkowski, and A. Skowron. "Rough sets and rough logic: A KDD perspective". In L. Polkowski, S. Tsumoto, and T.Y. Lin, editors, *Rough Set Methods and Applications*, pages 583–646. Physica-Verlag, 2000.

[49] L. Polkowski, S. Tsumoto, and T.Y. Lin, editors. *Rough Set Methods and Applications*. Physica-Verlag, 2000.

[50] H. Rasiowa and A. Skowron. "Approximation logic". In *Proceedings of Mathematical Methods of Speci£cation and Synthesis of Software Systems Conference*, pages 123–139. Akademie Verlag, 1985.

[51] H. Rasiowa and A. Skowron. "Rough concepts logic". In A. Skowron, editor, *Computation Theory*, LNCS 208, pages 288–297. Springer-Verlag, 1985.

[52] H. Rasiowa and A. Skowron. "The £rst step towards an approximation logic". *Journal of Symbolic Logic*, 51(2):509, 1986.

[53] P. Samarati. "Protecting respondents' identities in microdata release". *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.

[54] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report SRI-CSL-98-04, Computer Science Laboratory, SRI International, 1998.

[55] A. Skoworn and L. Polkowski, editors. *Rough Sets In Knowledge Discovery Vol 1: Methodology and Applications*. Physica-Verlag, 1998.

[56] A. Skoworn and L. Polkowski, editors. *Rough Sets In Knowledge Discovery Vol 2: Applications, Case Studies and Software Systems*. Physica-Verlag, 1998.

[57] L. Sweeney. "Guaranteeing anonymity when sharing medical data, the Data¤y system". In *Proceedings of American Medical Informatics Association*, 1997.

[58] D. Vakarelov. "Modal logics for knowledge representation". In A.R. Meyer and M.A. Taitslin, editors, *Proceedings of Symposium on Logical Foundations of Computer Science*, LNCS 363, pages 257–277. Springer-Verlag, 1989.

[59] D. Vakarelov. "A modal logic for similarity relations in Pawlak knowledge representation systems". *Fundamenta Informaticae*, 15:61–79, 1991.

[60] D. Vakarelov. "Modal logics for knowledge representation systems". *Theoretical Computer Science*, 90:433–456, 1991.

[61] D. Vakarelov. "Information systems, similarity, and modal logics". In E. Orłowska, editor, *Incomplete Information: Rough Set Analysis*, pages 492–550. Physica-Verlag, 1997.

[62] J. Wróblewski. "Analyzing relational databases using rough set based methods". In *Proceedings of the 8th International Conference on Processing and Management of Uncertainty in Knowledge-Based Systems*, volume 1, pages 256–262, 2000.

[63] Y.Y. Yao and C.J. Liau. "A generalized decision logic language for granular computing". In *Proceedings of the 11th IEEE International Conference on Fuzzy Systems*. IEEE Press, 2002.

[64] Y.Y. Yao and Q. Liu. "A generalized decision logic in interval-set-valued information tables". In N. Zhong, A. Skowron, and S. Ohsuga, editors, *New Directions in Rough Sets, Data Mining, and Granular-Soft Computing*, LNAI 1711, pages 285–293. Springer-Verlag, 1999.

[65] Y.Y. Yao and N. Zhong. "An analysis of quantitative measures associated with rules". In *Proceedings of the 2nd Paci£c-Asia Conference on Knowledge Discovery and Data Mining*, pages 479–488. IEEE Press, 1999.