Granular Computing as a Basis for Consistent Classification Problems

Y.Y. Yao, J.T. Yao
Department of Computer Science, University of Regina
Regina, Saskatchewan, Canada S4S 0A2
E-mail: {yyao, jtyao}@cs.uregina.ca

Abstract

Within a granular computing model of data mining, we reformulate the consistent classification problems. The granulation structures are partitions of a universe. A solution to a consistent classification problem is a definable partition. Such a solution can be obtained by searching a particular partition lattice. The new formulation enables us to precisely and concisely define many notions, and to present a more general framework for classification.

1 Introduction

As a recently renewed research topic, granular computing (GrC) is an umbrella term to cover any theories, methodologies, techniques, and tools that make use of granules (i.e., subsets of a universe) in problem solving [7, 9, 10]. Formal concept analysis may be considered as a concrete model of granular computing. It deals with the characterization of a concept by a unit of thoughts consisting of two parts, the intension and extension of the concept [1, 6]. The intension of a concept consists of all properties or attributes that are valid for all those objects to which the concept applies. The extension of a concept is the set of objects or entities which are instances of the concept.

Recently, a granular computing model for knowledge discovery and data mining is proposed by combining results from formal concept analysis and granular computing [8]. Each granule is viewed as the extension of a certain concept and a description of the granule is an intension of the concept. Knowledge discovery and data mining, especially rule mining, can be viewed as a process of forming concepts and finding relationships between concepts, in terms of their intensions and extensions.

The objective of this paper is to apply the granular computing model for the study of the consistent classification problems with respect to partitions of a universe. We reexpress, precisely and concisely, many notions based on partition lattice. Our reformulation of the consistent classification problems not only provides a formal treatment of the problem, but also brings more insights into the solution of the problem.

2 Formal Concept Analysis, Granular Computing, and Data Mining

A close connection between formal concept analysis, granular computing, and data mining can be established by focusing on their two fundamental and related tasks, namely, concept formation and concept relationship identification [8].

In the study of formal concepts, every concept is understood as a unit of thoughts that consists of two parts, the intension and extension of the concept [1, 6]. The intension (comprehension) of a concept consists of all properties or attributes that are valid for all those objects to which the concept applies. The extension of a concept is the set of objects or entities which are instances of the concept. All objects in the extension have the same properties that characterize the concept. In other words, the intension of a concept is an abstract description of common features or properties shared by elements in the extension, and the extension consists of concrete examples of the concept. A concept is thus described jointly by its intension and extension.

Basic ingredients of granular computing are subsets, classes, and clusters of a universe [7, 10]. There are many fundamental issues in granular computing, such as granulation of the universe, description of granules, relationships between granules, and computing with granules. Granulation of a universe involves the decomposition of the universe into parts, or the grouping of individual elements into classes, based on available information and knowledge. The construction of granules may be viewed as a process of identifying extensions of concepts. Similarly, finding a description of a granule may be viewed as searching for intension of a concept. Demri and Orlowska referred to such processes as the learning of extensions of concepts and learning of intensions of concepts, respectively [1]. The relationships between concepts, such as sub-concepts, disjoint and overlap concepts, and partial sub-concepts, can be inferred from intensions and extensions,

It may be argued that some of the main tasks of knowledge discovery and data mining are concept formation and concept relationship identification [2, 8, 9]. The results from formal concept analysis and granular computing can be immediately applied.

In a recently proposed granular computing model of data mining, it is assumed that information about a set of objects is given by an information table [8]. That is, an object is represented by a set of attribute-value pairs. A logic language is defined for the information table. The semantics of the language is given in the Tarski's style through the notions of a model and satisfiability. The model is an information table. An object satisfies a formula if the object has the properties as specified by the formula. Thus, the intension of a concept given by a formula of the language, and extension is given the set of objects satisfying the formula. This formulation enables us to study formal concepts in a logic setting in terms of intensions and also in a set-theoretic setting in terms of extensions.

In the following sections, we will demonstrate the application of the granular computing model for the study of a specific data mining problem known as the consistent classification problems.

3 Partition Lattices of Information Tables

In this section, we study the structures of several families of partitions.

3.1 Partition lattice

A partition provides a simple granulated view of a universe.

Definition 1 *A partition of a set U is a collection of nonempty, and pairwise disjoint subset of U whose union is U. The subsets in a partition are called blocks.*

When U is a finite set, a partition $\pi = \{X_i \mid 1 \leq i \leq m\}$ of U consists of a finite number of blocks. In this case, the conditions for partitions can be simply stated by:

$$\begin{array}{ll} \text{(i).} & \text{each } X_i \text{ is nonempty}, \\ \text{(ii).} & \text{for all } i \neq j, X_i \cap X_j = \emptyset, \\ \text{(iii).} & \bigcup \{X_i \mid 1 \leq i \leq m\} = U. \end{array}$$

There is a one-to-one correspondence between partitions of U and equivalence relations (i.e., reflexive, symmetric, and transitive relations) on U. Each equivalence class of the equivalence relation is a block of the corresponding partition. In this paper, we will use partitions and equivalence relations, and blocks and equivalence classes interchangeably.

Definition 2 *A partition π_1 is refinement of another partition π_2, or equivalently, π_2 is a coarsening of π_1, denoted by $\pi_1 \preceq \pi_2$, if every block of π_1 is contained in some block of π_2.*

The refinement relation is a partial ordering of the set of all partitions. Given two partitions π_1 and π_2, their meet, $\pi_1 \wedge \pi_2$, is the largest partition that is a refinement of both π_1 and π_2, their join, $\pi_1 \vee \pi_2$, is the smallest partition that is a coarsening of both π_1 and π_2. An equivalence classes of the meet are all nonempty intersections of an equivalence class from π_1 and an equivalence class from π_2, equivalence classes of the join are the smallest subsets which are exactly a union of equivalence classes from π_1 and π_2. Under these operations, the poset is a lattice called the partition lattice, denoted by $\Pi(U)$.

3.2 Information tables

An information table provides a convenient way to describe a finite set of objects called the universe by a finite set of attributes [4, 9].

Definition 3 *An information table is the following tuple:*

$$S = (U, At, \mathcal{L}, \{V_a \mid a \in At\}, \{I_a \mid a \in At\}),$$

where

U is a finite nonempty set of objects,
At is a finite nonempty set of attributes,
$\mathcal{L}$ is a language defined using attributes in At,
V_a is a nonempty set of values for $a \in At$,
$I_a : U \rightarrow V_a$ is an information function.

Each information function I_a is a total function that maps an object of U to exactly one value in V_a.

An information table represents all available information and knowledge. That is, objects are only perceived, observed, or measured by using a finite number of properties. We can easily extend the information function I_a to subsets of attribues. For a subset $A \subseteq At$, the value of an object x over A is denoted by $I_A(x)$.

Definition 4 *In the language $\mathcal{L}$, an atomic formula is given by $a = v$, where $a \in At$ and $v \in V_a$. If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, and $\phi \vee \psi$.*

The semantics of the language $\mathcal{L}$ can be defined in the Tarski's style through the notions of a model and satisfiability. The model is an information table S, which provides interpretation for symbols and formulas of $\mathcal{L}$.

Definition 5 *The satisfiability of a formula ϕ by an object x, written $x \models_S \phi$ or in short $x \models \phi$ if S is understood, is* *defined by the following conditions:*

$$
\begin{array}{ll}
(1) & x \models a = v \text{ iff } I_a(x) = v, \\
(2) & x \models \neg\phi \text{ iff } \text{ not } x \models \phi, \\
(3) & x \models \phi \wedge \psi \text{ iff } x \models \phi \text{ and } x \models \psi, \\
(4) & x \models \phi \vee \psi \text{ iff } x \models \phi \text{ or } x \models \psi.
\end{array}
$$

If ϕ is a formula, the set $m_S(\phi)$ defined by:

$$m_S(\phi) = \{x \in U \mid x \models \phi\}, \tag{1}$$

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

The meaning of a formula ϕ is therefore the set of all objects having the property expressed by the formula ϕ. In other words, ϕ can be viewed as the description of the set of objects $m(\phi)$. Thus, a connection between formulas of $\mathcal{L}$ and subsets of U is established.

With the introduction of language $\mathcal{L}$, we have a formal description of concepts. A concept definable in an information table is a pair $(\phi, m(\phi))$, where $\phi \in \mathcal{L}$. More specifically, ϕ is a description of $m(\phi)$ in S, the intension of concept $(\phi, m(\phi))$, and $m(\phi)$ is the set of objects satisfying ϕ, the extension of concept $(\phi, m(\phi))$.

To illustrate the idea developed so far, consider an information table given by Table 1, which is adopted from Quinlan [5]. The following expressions are some of the formulas of the language $\mathcal{L}$:

$$
\begin{array}{l}
\textbf{height} = \text{tall}, \\
\textbf{hair} = \text{dark}, \\
\textbf{height} = \text{tall} \wedge \textbf{hair} = \text{dark}, \\
\textbf{height} = \text{tall} \vee \textbf{hair} = \text{dark}.
\end{array}
$$

The meanings of the formulas are given by:

$$
\begin{array}{l}
m(\textbf{height} = \text{tall}) = \{o_3, o_4, o_5, o_6, o_7\}, \\
m(\textbf{hair} = \text{dark}) = \{o_4, o_5, o_7\}, \\
m(\textbf{height} = \text{tall} \wedge \textbf{hair} = \text{dark}) = \{o_4, o_5, o_7\}, \\
m(\textbf{height} = \text{tall} \vee \textbf{hair} = \text{dark}) = \{o_3, o_4, o_5, o_6, o_7\}.
\end{array}
$$

By pairing intensions and extensions, we can obtain formal concepts such as $(\textbf{height} = \text{tall}, \{o_3, o_4, o_5, o_6, o_7\})$ and $(\textbf{height} = \text{tall} \wedge \textbf{hair} = \text{dark}, \{o_4, o_5, o_7\})$.

3.3 Definable partition lattices

In an information table, some objects have the same description and hence can not be differentiated. With the indiscernibility of objects, a subset of objects may have to be considered as a whole rather than individuals. Consequently, for an arbitrary subset of the universe, $X \subseteq U$, it may be impossible to find a concept $(\phi, m(\phi))$ such that $m(\phi) = X$. For example, in Table 1, o_4 and o_5 have the same description. The subset $\{o_4, o_5\}$ must be considered a unit. It is impossible to find a formula whose meaning is $\{o_4, o_6, o_7\}$. In the case where we can precisely describe a subset of objects X, the description may not be unique. That is, there may exist two formulas such that $m(\phi) = m(\psi) = X$. For example, the two formulas:

Object	**height**	**hair**	**eyes**	**class**
o_1	short	blond	blue	+
o_2	short	blond	brown	-
o_3	tall	red	blue	+
o_4	tall	dark	blue	-
o_5	tall	dark	blue	-
o_6	tall	blond	blue	+
o_7	tall	dark	brown	-
o_8	short	blond	brown	-

Table 1: An information table

$$
\begin{array}{l}
\textbf{class} = +, \\
\textbf{hair} = \text{red} \vee (\textbf{hair} = \text{blond} \wedge \textbf{eyes} = \text{blue}),
\end{array}
$$

have the same meaning set $\{o_1, o_3, o_6\}$. Those observations lead us to consider only certain families of partitions from $\Pi(U)$.

Definition 6 *A subset $X \subseteq U$ is called a definable granule in an information table S if there exists at least one formula ϕ such that $m(\phi) = X$.*

Definition 7 *A partition π is called a definable partition in an information table S if every equivalence class is a definable granule.*

In information table 1, two objects o_4 and o_5, as well as o_2 and o_8, have the same description and are indistinguishable. Consequently, the smallest definable partition is $\{\{o_1\}, \{o_2, o_8\}, \{o_3\}, \{o_4, o_5\}, \{o_6\}, \{o_7\}\}$. The partition $\{\{o_1, o_2, o_3, o_4\}, \{o_5, o_6, o_7, o_8\}\}$ is not a definable partition.

If π_1 and π_2 are definable partitions, $\pi_1 \wedge \pi_2$ and $\pi_1 \vee \pi_2$ are definable partitions. The set of all definable partitions $\Pi_{\mathrm{D}}(U)$ is a sub-lattice of $\Pi(U)$.

In many machine learning algorithms, one is only interested in formulas of certain form. Suppose we restrict the connectives of language $\mathcal{L}$ to only the conjunction connective $\wedge$. Each formula is a conjunction of atomic formulas and such a formula is referred to as a conjunctor.

Definition 8 *A subset $X \subseteq U$ is a conjunctively definable granule in an information table S if there exists a conjunc-*

tor ϕ such that $m(\phi) = X$. A partition π is called a conjunctively definable partition if every equivalence class is a conjunctively definable granule.

The partition $\{\{o_1, o_2, o_6, o_8\}, \{o_3, o_4, o_5, o_7\}\}$ is a definable partition but not a conjunctively definable partition in the information table 1. The join of two conjunctively definable partitions $\{\{o_1, o_2, o_8\}, \{o_3, o_4, o_5, o_6, o_7\}\}$ and $\{\{o_1, o_2, o_6, o_8\}, \{o_3\}, \{o_4, o_5, o_7\}\}$ is $\{U\}$, which is not a conjunctively definable partition.

The meet, $\pi_1 \wedge \pi_2$, of two conjunctively definable partitions is a conjunctively definable partition. However, the join, $\pi_1 \vee \pi_2$, is not necessarily a conjunctively definable partition. In this case, we only obtain a meet semi-lattice $\Pi_{\mathrm{CD}}(U)$.

A lattice related to $\Pi_{\mathrm{CD}}(U)$ is the lattice formed by partitions defined by various subsets of At. For a subset of attributes A, we can define an equivalence relation E_A as follows:

$$\begin{aligned} xE_Ay &\iff \text{for all } a \in A, I_a(x) = I_a(y) \\ &\iff I_A(x) = I_A(y). \end{aligned} \tag{2}$$

For the empty set, we obtain the coarsest partition $\{U\}$. For a nonempty subset of attributes, the induced partition is conjunctively definable. The family of partition defined by subsets of attributes form a lattice $\Pi_{\mathrm{AD}}(U)$, which is not necessarily a sub-lattice of $\Pi(U)$.

For the information table 1, we obtain the following partitions with respect to subsets of the attributes.

$$\begin{aligned}
\pi_0 : \quad & \emptyset, \\
& \{U\}, \\
\pi_1 : \quad & \{\mathbf{height}\}, \\
& \{\{o_1, o_2, o_8\}, \{o_3, o_4, o_5, o_6, o_7\}\}, \\
\pi_2 : \quad & \{\mathbf{hair}\}, \\
& \{\{o_1, o_2, o_6, o_8\}, \{o_3\}, \{o_4, o_5, o_7\}\}, \\
\pi_3 : \quad & \{\mathbf{eyes}\}, \\
& \{\{o_1, o_3, o_4, o_5, o_6\}, \{o_2, o_7, o_8\}\}, \\
\pi_4 : \quad & \{\mathbf{height}, \mathbf{hair}\}, \\
& \{\{o_1, o_2, o_8\}, \{o_3\}, \{o_4, o_5, o_7\}, \{o_6\}\}, \\
\pi_5 : \quad & \{\mathbf{height}, \mathbf{eyes}\}, \\
& \{\{o_1\}, \{o_2, o_8\}, \{o_3, o_4, o_5, o_6\}, \{o_7\}\}, \\
\pi_6 : \quad & \{\mathbf{hair}, \mathbf{eyes}\}, \\
& \{\{o_1, o_6\}, \{o_2, o_8\}, \{o_3\}, \{o_4, o_5\}, \{o_7\}\}, \\
\pi_7 : \quad & \{\mathbf{height}, \mathbf{hair}, \mathbf{eyes}\}, \\
& \{\{o_1\}, \{o_2, o_8\}, \{o_3\}, \{o_4, o_5\}, \{o_6\}, \{o_7\}\}.
\end{aligned}$$

Since each subset defines a different partition, the partition lattice has the same structure as the lattice defined by the power set of the three attributes $\mathbf{height}$, $\mathbf{hair}$, and $\mathbf{eyes}$.

All the notions developed in this section can be defined relative to a particular subset $A \subseteq At$ of attributes. A subset $X \subseteq U$ is called a definable granule with respect to a subset of attributes $A \subseteq At$ if there exists a least one formula ϕ over A such that $m(\phi) = X$. A partition π is called a definable partition with respect to a subset of attributes A if every equivalence class is a definable granule with respect to A. Let $\Pi_{\mathrm{D}(A)}(U)$, $\Pi_{\mathrm{CD}(A)}(U)$, and $\Pi_{\mathrm{AD}(A)}(U)$ denote the partition (semi-) lattices with respect to a subset of attributes $A \subseteq At$, respectively. We have the following connection between partition (semi-) lattices:

$$\begin{aligned}
&\Pi_{\mathrm{AD}}(U) \subseteq \Pi_{\mathrm{CD}}(U) \cup \{U\} \subseteq \Pi_{\mathrm{D}}(U) \subseteq \Pi(U), \\
&\Pi_{\mathrm{AD}(A)}(U) \subseteq \Pi_{\mathrm{CD}(A)}(U) \cup \{U\} \subseteq \Pi_{\mathrm{D}(A)}(U) \subseteq \Pi(U).
\end{aligned}$$

They provide a formal framework of classification problems.

4 Classification as Partition Lattice Search

Classification problem is one of the well studied problems in machine learning and data mining. In this section, we reformulate the classification problem using partition lattice.

4.1 Formulation of the problem

In supervised classification, it is assumed that each object is associated with a unique class label. Objects are divided into disjoint classes which form a partition of the universe. We further assume that information about objects are given by an information table. Without loss of generality, we assume that there is a unique attribute $\mathbf{class}$ taking class labels as its value. The set of attributes is expressed as $At = C \cup \{\mathbf{class}\}$, where C is the set of attributes used to describe the objects. The goal is to find classification rules of the form, $\phi \Longrightarrow \mathbf{class} = c_i$, where ϕ is a formula over C and c_i is a class label.

Let $\pi_{\mathbf{class}} \in \Pi(U)$ denote the partition induced by the attribute $\mathbf{class}$. An information table with a set of attributes $At = C \cup \{\mathbf{class}\}$ is said to provide a consistent classification if all objects with the same description over C have the same class label, namely, if $I_C(x) = I_C(y)$, then $I_{\mathbf{class}}(x) = I_{\mathbf{class}}(y)$. Using the concept of partition lattice, we immediately have the equivalent definition.

Definition 9 *An information table with a set of attributes $At = C \cup \{\mathbf{class}\}$ is a consistent classification problem if and only if there exists a partition $\pi \in \Pi_{\mathrm{AD}(C)}(U)$ such that $\pi \preceq \pi_{\mathbf{class}}$.*

In the rest of this paper, we restrict our discussion to the consistent classification problem.

Definition 10 *The solution to a consistent classification problem is a definable partition π such that $\pi \preceq \pi_{\mathbf{class}}$. For a pair of equivalence classes $X \in \pi$ and $Y \in \pi_{\mathbf{class}}$ with $X \subseteq Y$, we can derive a classification rule $\phi(X) \Longrightarrow \phi(Y)$, where $\phi(X)$ and $\phi(Y)$ are the formulas whose meaning sets are X and Y, respectively.*

For the information table 1, the definable partition,

$$\{\{o_1, o_6\}, \{o_2, o_7, o_8\}, \{o_3\}, \{o_4, o_5\}\} \preceq \\ \{\{o_1, o_3, o_6\}, \{o_2, o_4, o_5, o_7, o_8\}\} = \pi_{\mathbf{class}},$$

is a solution to the classification problem. The classification rules corresponding to the solution are given by:

$$\begin{aligned} &\mathbf{hair} = \text{blond} \wedge \mathbf{eyes} = \text{blue} \Longrightarrow \mathbf{class} = +, \\ &\mathbf{eyes} = \text{brown} \Longrightarrow \mathbf{class} = -, \\ &\mathbf{hair} = \text{red} \Longrightarrow \mathbf{class} = +, \\ &\mathbf{hair} = \text{dark} \wedge \mathbf{eyes} = \text{blue} \Longrightarrow \mathbf{class} = -. \end{aligned}$$

The left hand side of a rule is a formula whose meaning is a block of the solution partition. For example, for the first rule, we have $m(\mathbf{hair} = \text{blond} \wedge \mathbf{eyes} = \text{blue}) = \{o_1, o_6\}$.

For a consistent classification problem, the partition defined by all attributes in C is the smallest partition in the three definable partition lattices. Let π_A denote the partition defined by a subset $A \subseteq C$ of attributes. The smallest partition π_C is a trivial solution to the consistent classification problem.

Depending on the particular partition lattice used, one can easily establish properties of the family of solution partitions. Let $\Pi_\alpha(U)$, where $\alpha = \mathrm{AD}(C), \mathrm{CD}(C), \mathrm{D}(C)$, denote a (semi-) lattice of definable partitions. Let $\Pi_\alpha^S(U)$ be the corresponding set of all solution partitions. We have:

(i). For $\alpha = \mathrm{AD}(C), \mathrm{CD}(C), \mathrm{D}(C)$, if $\pi' \in \Pi_\alpha(U)$, $\pi \in \Pi_\alpha^S(U)$ and $\pi' \preceq \pi$, then $\pi' \in \Pi_\alpha^S(U)$;

(ii). For $\alpha = \mathrm{AD}(C), \mathrm{CD}(C), \mathrm{D}(C)$, if $\pi', \pi \in \Pi_\alpha^S(U)$, then $\pi' \wedge \pi \in \Pi_\alpha^S(U)$;

(iii). For $\alpha = \mathrm{D}(C)$, if $\pi', \pi \in \Pi_\alpha^S(U)$, then $\pi' \vee \pi \in \Pi_\alpha^S(U)$;

It follows that the set of all solution partitions form a lattice or meet semi-lattice.

Mining classification rules can be formulated as a search for a partition from a partition lattice. A definable lattice provides the search space of potential solutions, and the partial order of the lattice provides the search direction. The standard search methods, such as depth-first search, breadth-first search, bounded depth-first search, and heuristic search, can be used to find a solution from a lattice of definable partitions. Depending on the required properties of rules, one may use different definable partition lattice introduced earlier. For example, by search the semi-lattice $\Pi_{\mathrm{CD}(C)}(U)$, we can obtain classification rules whose left hand sides are only conjunction of atomic formulas. The well known ID3 learning algorithm in fact searches $\Pi_{CD(C)}(U)$ for classification rules [5]. By searching the lattice $\Pi_{\mathrm{AD}(C)}(U)$, one can obtain a similar solution.

We can re-express many fundamental notions of classification in terms of partitions.

Definition 11 *For two solutions $\pi_1, \pi_2 \in \Pi_\alpha$ of a consistent classification problem, namely, $\pi_1 \preceq \pi_{\mathbf{class}}$ and $\pi_2 \preceq \pi_{\mathbf{class}}$, if $\pi_1 \preceq \pi_2$, we say that π_1 is a more specific solution than π_2, or equivalently, π_2 is a more general solution that π_1.*

Definition 12 *A solution $\pi \in \Pi_\alpha$ of a consistent classification problem is called the most general solution if there does not exists another solution $\pi' \in \Pi_\alpha$, $\pi \neq \pi'$, such such that $\pi \preceq \pi' \preceq \pi_{\mathbf{class}}$.*

In the information table 1, consider three partitions:

$$\begin{aligned} \pi_1: &\quad \{\{o_1\}, \{o_2, o_8\}, \{o_3\}, \{o_4, o_5\}, \{o_6\}, \{o_7\}\}, \\ \pi_2: &\quad \{\{o_1, o_6\}, \{o_2, o_8\}, \{o_3\}, \{o_4, o_5, o_7\}\}, \\ \pi_3: &\quad \{\{o_1, o_6\}, \{o_2, o_7, o_8\}, \{o_3\}, \{o_4, o_5\}\}. \end{aligned}$$

from the lattice $\Pi_{\mathrm{CD}(C)}(U)$. We have $\pi_1 \preceq \pi_2 \preceq \pi_{\mathbf{class}}$ and $\pi_1 \preceq \pi_3 \preceq \pi_{\mathbf{class}}$. Thus, π_1 is a more specific solution than both π_2 and π_3. In fact, π_2 and π_3 are two most general solutions.

For a consistent classification problem, the partition defined by all attributes in C is the smallest partition in Π_α. Thus, a most general solution always exists. However, a most general solution may not be unique. There may exist many more general solutions.

The roles played by each attribute, well studied in the theory of rough sets [4], can be re-expressed as follows.

Definition 13 *An attribute $a \in C$ is called a core attribute if $\pi_{C-\{a\}}$ is not a solution to the consistent classification problem.*

Definition 14 *An attribute $a \in C$ is called a superfluous attribute if $\pi_{C-\{a\}}$ is a solution to the consistent classification problem, namely, $\pi_{C-\{a\}} \preceq \pi_{\mathbf{class}}$.*

Definition 15 *A subset $A \subseteq C$ is called a reduct if π_A is a solution to the consistent classification problem and π_B is not a solution for any proper subset $B \subset A$.*

For a given consistent classification problem, there may exist more than one reduct.

In the information table 1, attributes $\mathbf{hair}$ and $\mathbf{eyes}$ are core attributes. Attribute $\mathbf{height}$ is a superfluous attribute. The only reduct is the set of attributes $\{\mathbf{hair}, \mathbf{eyes}\}$.

4.2 ID3 type search algorithms

ID3 type learning algorithms can be formulated as a heuristic search of the semi-lattice $\Pi_{\mathrm{CD}(C)}(U)$. The heuristic used is based on an information-theoretic measure of dependency between the partition defined by $\mathbf{class}$ and another conjunctively definable partition with respect to the set of attributes C. Roughly speaking, the measure quantifies the degree to which a partition $\pi \in \Pi_{\mathrm{CD}(C)}(U)$ satisfies the condition $\pi \preceq \pi_{\mathbf{class}}$ of a solution partition.

Specifically, the direction of ID3 search is from coarsest partitions of $\Pi_{\mathrm{CD}(C)}(U)$ to more refined partitions. Largest partitions in $\Pi_{\mathrm{CD}(C)}(U)$ are the partitions defined by single attributes in C. Using the information-theoretic measure, ID3 first selects a partition defined by a single attribute. If an equivalence class in the partition is not a conjunctively definable granule with respect to $\mathbf{class}$, the equivalence class is further divided into smaller granules by using an additional attribute. The same information-theoretic measure is used for the selection of the new attribute. The smaller granules are conjunctively definable granules with respect to C. The search process continues until a partition $\pi \in \Pi_{\mathrm{CD}(C)}(U)$ is obtained such that $\pi \preceq \pi_{\mathbf{class}}$.

4.3 Rough set type search algorithms

Algorithms for finding a reduct in the theory of rough sets can be viewed as heuristic search of the partition lattice $\Pi_{\mathrm{AD}(C)}(U)$. Two directions of search can be carried, either from coarsening partitions to refinement partitions or from refinement partitions to coarsening partitions.

The smallest partition in $\Pi_{\mathrm{AD}(C)}(U)$ is π_C. By dropping an attribute a from C, one obtains a coarsening partition $\pi_{C-\{a\}}$. Typically, a certain fitness measure is used for the selection of the attribute. The process continues until no further attributes can be dropped. That is, we find a subset $A \subseteq C$ such that $\pi_A \preceq \pi_{\mathbf{class}}$ and $\neg(\pi_B \preceq \pi_{\mathbf{class}})$ for all proper subsets $B \subset A$. The resulting set of attributes A is a reduct.

The largest partition in $\Pi_{\mathrm{AD}(C)}(U)$ is $\pi_\emptyset$. By adding an attribute a, one obtains a refined partition π_a. The process continues until we have a partition satisfying the condition $\pi_A \preceq \pi_{\mathbf{class}}$. The resulting set of attributes A is a reduct.

5 Conclusion

The granular computing model for data mining is used to reformulate the consistent classification problems. We explore the structures of partitions of a universe. The consistent classification problems are expressed as the relationships between partitions of the universe. Three definable partition lattices are introduced. Depending on the properties of classification rules, a solution to a consistent classification problem is a definable partition in one of the lattices. Such a solution can be obtained by searching that lattice. Our formulation is similar to the well established version space search method for machine learning [3].

The new formulation enables us to precisely and concisely define many notions, and to present a more general framework for classification. To illustrate its the potential usefulness and generality, we briefly describe the ID3 and rough set learning algorithms using the proposed model.

References

[1] Demri, S. and Orlowska, E. Logical analysis of indiscernibility, in: *Incomplete Information: Rough Set Analysis*, Orlowska, E. (Ed.), Physica-Verlag, Heidelberg, pp. 347-380, 1998.

[2] Fayyad, U.M. and Piatetsky-Shapiro, G. (Eds.) *Advances in Knowledge Discovery and Data Mining*, AAAI Press, 1996.

[3] Mitchell, T.M. Generalization as search, *Artificial Intelligence*, **18**, 203-226, 1982.

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

[5] Quinlan, J.R. Learning efficient classification procedures and their application to chess end-games, in: *Machine Learning: An Artificial Intelligence Approach*, Vol. 1, Michalski, J.S., Carbonell, J.G., and Mirchell, T.M. (Eds.), Morgan Kaufmann, Palo Alto, CA, pp. 463-482, 1983.

[6] Wille, R. Concept lattices and conceptual knowledge systems, *Computers Mathematics with Applications*, **23**, 493-515, 1992.

[7] Yao, Y.Y. Granular computing: basic issues and possible solutions, *Proceedings of the 5th Joint Conference on Information Sciences*, pp.186-189, 2000.

[8] Yao, Y.Y. On modeling data mining with granular computing, *Proceedings of COMPSAC 2001*, pp.638-643, 2001.

[9] Yao, Y.Y. and Zhong, N. Potential applications of granular computing in knowledge discovery and data mining, *Proceedings of World Multiconference on Systemics, Cybernetics and Informatics*, pp.573-580, 1999.

[10] Zadeh, L.A. Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, *Fuzzy Sets and Systems*, **19**, 111-127, 1997.