Definability in Logic and Rough Set Theory [1]

Tuan-Fang Fan[2] and **Churn-Jung Liau**[3] and **Duen-Ren Liu**[4]

Abstract. Rough set theory is an effective tool for data mining. According to the theory, a concept is definable if it can be written as a Boolean combination of equivalence classes induced from classification attributes. On the other hand, definability in logic has been explicated by Beth's theorem. In this paper, we propose two data representation formalisms, called first-order data logic (FODL) and attribute value-sorted logic (AVSL), respectively. Based on these logics, we explore the relationship between logical definability and rough set definability.

1 Introduction

In recent years, knowledge discovery in databases (KDD) and data mining have received more and more attention because of their practical applications. The rough set theory proposed by Pawlak provides an effective tool for extracting knowledge from data tables [3]. To represent and reason about the extracted knowledge, a decision logic (DL) is also proposed in [3]. The semantics of the logic is defined in a Tarskian style through the notions of models and satisfaction. While DL is an instance of propositional logic, we can also represent knowledge in data tables by using first-order logic (FOL)[2] or many-sorted first-order logic (MSFOL). In this paper, we investigate the definability of concepts in the context of these alternative logical descriptions of data tables. In the next section, we review rough set theory, with the emphasis on the notion of definability. Then, in Sections 3 and 4, we propose first-order data logic and attribute value-sorted logic for the description of data tables respectively, and discuss the relationship between logical definability and rough set definability in the context of these logics. We conclude the paper in Section 5.

2 Rough Set Theory—A Review

The basic construct of rough set theory is an *approximation space*, which is defined as a pair (U, R), where U is the universe and $R \subseteq U \times U$ is an equivalence relation on U. We can write an equivalence class of R as $[x]_R$ if it contains the element x. Note that $[x]_R = [y]_R$ iff $(x, y) \in R$.

In philosophy, the extension of a concept is defined as the objects that are instances of the concept. Following the terminology, a subset of the universe is called a *concept* or a *category* in rough set theory. Given an approximation space (U, R), each equivalence class of R is called an *R-basic category* or *R-basic concept*, and any union of R-basic categories is called an R-category. Now, for an arbitrary concept $X \subseteq U$, we are interested in the definability of X by using R-basic categories. We say that X is *R-definable* if X is an R-category; otherwise X is *R-undefinable*. The R-definable concepts are also called *R-exact* sets, whereas R-undefinable concepts are said to be *R-inexact* or *R-rough*. A rough set can be approximated by two exact sets, called the lower approximation and upper approximation of X, respectively, and defined as follows:

$$\underline{R}X = \{x \in U \mid [x]_R \subseteq X\},$$

$$\overline{R}X = \{x \in U \mid [x]_R \cap X \neq \emptyset\}.$$

Obviously, a set X is R-definable iff $\underline{R}X = \overline{R}X$.

In data mining problems, the equivalence relation is determined by the attributes (features) used to classify objects. Two objects are equivalent if they have the same values in every such attribute. Thus, intuitively, a concept is definable in rough set theory if it can be precisely described by such attributes.

3 Definability in First-order Data Logic

To describe data tables by (a fragment of) FOL, we use an instance of function-free monadic predicate logic, called *first-order data logic* (FODL). The alphabet (or vocabulary) of FODL consists of a set of constant symbols, a *finite* set of monadic predicate symbols, a set of variables, Boolean connectives $(\neg, \wedge, \vee, \supset, \equiv)$, and the quantifiers $(\forall, \exists)$. The syntax and semantics of FODL are the same as those of ordinary FOL[2].

Based on FODL, we can formulate the definability of a concept in rough set theory precisely. In the language of FODL, a concept corresponds to a predicate, and the equivalence relation in an approximation space can be determined by a set of predicates. Let $\mathbf{S}$ be a subset of predicates. Then the following formula defines an indiscernibility relation (with respect to $\mathbf{S}$):

$$\eta_{\mathbf{S}}(x, y) = \bigwedge_{P \in \mathbf{S}} P(x) \equiv P(y).$$

Given an arbitrary predicate P, we can define two formulas corresponding to the lower and upper approximations of P:

$$\underline{P_{\mathbf{S}}}(x) = \forall y(\eta_{\mathbf{S}}(x, y) \supset P(y)),$$

$$\overline{P_{\mathbf{S}}}(x) = \exists y(\eta_{\mathbf{S}}(x, y) \wedge P(y)).$$

Let Γ be an FODL theory that contains only predicate symbols in $\mathbf{S} \cup \{P\}$. Then we say that P is $\mathbf{S}$-*definable* with respect to Γ if

$$\Gamma \models \forall x(\underline{P_{\mathbf{S}}}(x) \equiv \overline{P_{\mathbf{S}}}(x)),$$

where $\models$ means the semantic consequence relation in FODL.

[1] This work was partially supported by NSC (Taiwan) under grants 95-2221-E-001-029-MY3

[2] Department of Computer Science and Information Engineering, National Penghu University, Penghu 880, Taiwan, email:dffan@npu.edu.tw, and Institute of Information Management, National Chiao-Tung University, Hsinchu 300, Taiwan, email: tffan.iim92g@nctu.edu.tw

[3] Institute of Information Science, Academia Sinica, Taipei 115, Taiwan, email: liaucj@iis.sinica.edu.tw

[4] Institute of Information Management, National Chiao-Tung University, Hsinchu 300, Taiwan, email: dliu@iim.nctu.edu.tw

In classical logic, the definability of a predicate is explicated by the well-known Beth's definability theorem[1]. The theorem states that explicit definability is equivalent to implicit definability. Let Γ be an FODL theory that contains only predicate symbols in $\mathbf{S} \cup \{P\}$. Then Γ *explicitly defines* P if there exists a wff $\varphi(x)$ that contains only predicate symbols in $\mathbf{S}$ such that

$$\Gamma \models \forall x(\varphi(x) \equiv P(x)).$$

We say that Γ *implicitly defines* P if for any $\mathfrak{A}, \mathfrak{B} \in Mod(\Gamma)$ such that $Q^{\mathfrak{A}} = Q^{\mathfrak{B}}$ for all $Q \in \mathbf{S}$, we have $P^{\mathfrak{A}} = P^{\mathfrak{B}}$, where $Mod(\Gamma)$ is the set of models of Γ. In effect, the implicit definability of a predicate P means the possibility of uniquely characterizing P. The primary objective of this paper is to establish the relationship between logical definability and rough set definability.

Theorem 1 *Let Γ be an FODL theory that contains only predicate symbols in $\mathbf{S} \cup \{P\}$. Then the explicit (or implicit) definability of P in Γ implies that P is $\mathbf{S}$-definable with respect to Γ.*

4 Definability in Attribute Value-sorted Logic

In FODL, a monadic predicate intuitively corresponds to an attribute-value pair. However, in many cases, the number of possible values for an attribute may be infinite. In such infinite-domain cases, an infinite number of predicates must be available in FODL, but since the indiscernibility wff $\eta_{\mathbf{s}}$ can only be defined with respect to a finite subset of predicates $\mathbf{S}$, it is sometimes inadequate. To circumvent such difficulties, we can use many-sorted first-order logic (MSFOL) as the data representation formalism.

4.1 Syntax and semantics

We use a special instance of MSFOL, called *attribute value-sorted logic* (AVSL), to describe data tables. The set of sorts for AVSL is $\Sigma = \{\sigma_i \mid i \in I\} \cup \{\sigma_u\}$, where I is an index set. The sort σ_u is called the *object sort* and each σ_i is called an *attribute value sort*.

As in the case of FODL, the alphabet (or vocabulary) of AVSL consists of constant symbols, predicate symbols, variables, and logical symbols. The only difference is that, in AVSL, a rank function is used to assign a *rank* to constant symbols, predicate symbols, and variables. The rank of a constant symbol or a variable is an element of Σ, and the rank of a predicate symbol is in Σ^k if its arity is k. A constant (resp. variable) of rank σ_u is called an *object constant* (resp. *variable*); otherwise, it is called an *attribute domain constant* (resp. *variable*). We assume that the set of predicate symbols is the union of a set of monadic predicates and the set of dyadic predicates $\{R_i \mid i \in I\}$. For each $i \in I$, R_i is of rank (σ_u, σ_i), and called an *attribute predicate*. Also, a monadic predicate of rank σ_u is called a *concept predicate*; and for each $i \in I$, a monadic predicate of rank σ_i is called a *value predicate*. Now, a term is either a constant or a variable, and the rank of the term is that of the constant or variable. If P is a predicate of rank $(\sigma_1, \cdots, \sigma_k)$ and $t_1, t_2, \cdots, t_k$ are terms of ranks $\sigma_1, \sigma_2, \cdots, \sigma_k$ respectively, then $P(t_1, t_2, \cdots, t_k)$ is an atomic formula ($k = 1, 2$). The formation rules for compound wffs are the same as those for ordinary FOL[2].

4.2 Logical definability

Analogous to the case of FODL, we can formulate the definability of a rough concept in AVSL. Let x and y be object variables, v be an attribute domain variable, and $\mathbf{S}$ be a subset of the index set I. Then we can define the indiscernibility formula (with respect to $\mathbf{S}$) as:

$$\varepsilon_{\mathbf{s}}(x, y) = \bigwedge_{i \in \mathbf{s}} \forall v(R_i(x, v) \equiv R_i(y, v)).$$

Again, given an arbitrary concept predicate P, we can define two formulas corresponding to its lower and upper approximations:

$$\underline{\varepsilon} P_{\mathbf{s}}(x) = \forall y(\varepsilon_{\mathbf{s}}(x, y) \supset P(y)),$$
$$\overline{\varepsilon} P_{\mathbf{s}}(x) = \exists y(\varepsilon_{\mathbf{s}}(x, y) \wedge P(y)).$$

Let Γ be an AVSL theory that contains only predicate symbols in $\{R_i \mid i \in \mathbf{S}\} \cup \{P\}$. Then we say that P is *indiscernibly* $\mathbf{S}$*-definable* with respect to Γ if

$$\Gamma \models \forall x(\underline{\varepsilon} P_{\mathbf{s}}(x) \equiv \overline{\varepsilon} P_{\mathbf{s}}(x)).$$

The definition of the explicit and implicit definability of P in Γ is the same as that in the FODL case and, analogously, we have the following theorem.

Theorem 2 *Let Γ be an AVSL theory that contains only predicate symbols in $\{R_i \mid i \in \mathbf{S}\} \cup \{P\}$. Then the explicit definability of P in Γ implies that P is indiscernibly $\mathbf{S}$-definable with respect to Γ.*

In addition to Pawlak's approximation space, the notion of *tolerance approximation spaces* has been proposed in [4] to cope with the problem of imprecise boundary regions in rough set theory. The definability of a concept in a tolerance approximation space can also be formulated in AVSL. First, let x, y, v and $\mathbf{S}$ be defined as above. Then the tolerance formula (with respect to $\mathbf{S}$) is

$$\tau_{\mathbf{s}}(x, y) = \bigwedge_{i \in \mathbf{s}} \exists v(R_i(x, v) \wedge R_i(y, v)).$$

Second, the lower and upper approximations of a concept predicate P are defined as follows:

$$\underline{\tau} P_{\mathbf{s}}(x) = \forall y(\tau_{\mathbf{s}}(x, y) \supset P(y)),$$
$$\overline{\tau} P_{\mathbf{s}}(x) = \exists y(\tau_{\mathbf{s}}(x, y) \wedge P(y)).$$

Finally, let Γ be an AVSL theory that contains only predicate symbols in $\mathbf{S} \cup \{P\}$ such that $\{\bigwedge_{i \in \mathbf{s}} \forall x \exists v R_i(x, v)\} \subseteq \Gamma$. Then we say that P is *tolerantly* $\mathbf{S}$*-definable* with respect to Γ if

$$\Gamma \models \forall x(\underline{\tau} P_{\mathbf{s}}(x) \equiv \overline{\tau} P_{\mathbf{s}}(x)).$$

Note that, to ensure the reflexivity of the tolerance relation, $\forall x \exists v R_i(x, v)$ is included in Γ for each $i \in \mathbf{S}$. However, logical definability no longer implies rough set definability in terms of the tolerance approximation space.

5 Conclusion

In this paper, we propose using FODL and AVSL for logical descriptions of data tables. Based on these logics, we precisely formulate the notion of definability in rough set theory and discuss its relationship to explicit and implicit definability in classical logic.

REFERENCES

[1] E.W. Beth. On padoa's method in the theory of definition. *Indagationes Math.*, 15:330–339, 1953.

[2] E. Mendelson. *Introduction to Mathematical Logic*. Chapman & Hall/CRC, forth edition, 1997.

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

[4] A. Skowron and J. Stepaniuk. Tolerance approximation spaces. *Fundamenta Informaticae*, 27(2/3):245–253, 1996.