

ating system deems appropriate, that is, only when an acceptable key is pressed. So the right display message ought to be, "press any acceptable key to continue." However, such an instruction would be meaningless, because the term "acceptable" has no obvious interpretation.

The moral? We as programmers should take into account alternative perspectives, especially

the users' perspective. To understand a communication from the perspective of its recipient is one of the lessons of deconstruction theory.

A lot of people talk of the importance of being user-centered. Here is a case where being user-centered has direct consequences on our programs. Although thinking from another perspective is rarely as easy as in

the case of the "any" key, we must keep trying.

As for the matter of acknowledged continuation, just ask the user to press the space bar. **C**

MUNINDAR P. SINGH (singh@ncsu.edu) teaches computer science at North Carolina State University in Raleigh.

MONA SINGH (mona.singh@ericsson.com) is with the Ericsson New Concepts group in Research Triangle Park, NC.

© 2000 ACM 0002-0782/00/0400 \$5.00

Jerzy W. Grzymala-Busse and Wojciech Ziarko

Data Mining and Rough Set Theory



This is in response to "Myths about Rough Set Theory" (Nov. 1998, p. 102) by W.W. Koczkodaj, M. Orlowski, and V.W. Marek. The authors raise some important issues and express some legitimate concerns. We are surprised they list rough set theory as the only discipline in which there are two of the cited problems—the discipline in which discretization is necessary or which deals with complex data. The third problem raised by the authors is associated with the difference between objective and subjective approaches to uncertainty.

Let us start with discretization. Many people deal with discretization unknowingly. For example, in grading student work, there are usual cut-points (90% for an "A," 80% for a "B," and so forth); original scores are replaced by intervals, coded by "A," "B," and so on. The authors are probably confused by the fact that in some

applications of rough set theory, discretization is used as a preprocessing. However, discretization is required in all rule (or tree) induction systems. Such systems constitute the core of data mining (or knowledge discovery). Many such well-known systems (such as C4.5, based on conditional entropy or CART, based on Bayes rule) are equipped with their own discretization schemes. Neither C4.5 nor CART use rough set theory. Practically every machine learning system uses discretization while very few of them are based on rough set theory. To complicate matters, discretization methods used in rule induction systems based on rough set theory, such as KDD-R or LERS, are not based on rough set theory (for example, both KDD-R and LERS use statistical methods). Furthermore, these discretization methods could be used in other systems, (in C4.5 or CART). On the other hand, all

existing discretization methods, based on many different approaches to uncertainty, could be used as preprocessing for KDD-R and LERS. Discretization is a technique used in many areas, including machine learning and learning in Bayesian networks, and is definitely not restricted to rule induction systems based on rough set theory.

To illustrate the complexities involved in data analysis, the authors refer to an example of a table with 10 attributes, each with 20 values, which is likely to lead to a large "number of possible instances." However, one could cite this kind of example to illustrate potential problems occurring in all disciplines dealing with data, starting from statistics, through database management, and ending with machine learning (for the sake of correctness, it is not clear what the authors mean by "the number of possible instances."

Most likely they refer to the number of possible different cases (rows) of the table. They are mistaken. The correct number is $2010^{10} = 1.024 \cdot 10^{13}$. In all of these areas we may deal with big data sets and with potentially large number of different data sets. Again, the problem is common to all of these disciplines and by no means occurs just in rough set theory. Fortunately, rough set theory offers algorithms with polynomial time complexity and space complexity with respect to the number of attributes and the number of cases.

Finally, regarding the authors' comments about objectivity of rough set theory, we are puzzled why they assume objectivity means superiority. They confuse daily life with science. In real life we would like to have objective managers, for example, giving salary raises based on merit. But this is not how the subjective or objective approach is understood in science. Probability theory is a good example. It is an example of

well-established calculus of uncertainty. For a long time there was a dispute (and still is) between the objective approach to the definition of probability (based on experiments and relative frequencies of outcomes) and the subjective approach (based on experts' opinions). For example, an individual may observe a game based on a random process and evaluate probabilities. This is the objective approach. Or, the individual may ask a gambler how he or she will bet his or her own money. This is the subjective approach. Currently, subjectivism prevails in probability theory. The proponents of the subjective approach do not show any inferiority complex. The problem is definitely not which approach is superior. The Dempster-Shafer theory uses the subjective approach to uncertainty because its fundamental tool, a belief function, should be estimated by experts. There are close relations between the Dempster-Shafer theory and the rough set theory. Both theories describe

the same phenomena. However, in rough set theory the basic tools are sets: lower and upper approximations of the concept. These sets are well defined and are computed directly from the input data. Thus, rough set theory is objective, but it does not mean that it is superior (or inferior). For example, if input data were preprocessed and numerical attributes were discretized by an expert, the resulting data might be subjective. But again, this preprocessing is not a part of rough set theory, as we explained previously. Input data must be given to initiate rough set theory procedures, and, when rough set theory comes into the picture, its methods are objective with respect to given data. **□**

JERZY W. GRZYMALA-BUSSE

(jerzy@eecs.ukans.edu) is a professor in the Department of Electrical Engineering and Computer Science at the University of Kansas.

WOJCIECH ZIARKO (ziarko@cs.uregina.ca) is a professor in the Computer Science department at the University of Regina, Canada.

© 2000 ACM 0002-0782/00/0400 \$5.00

Friedrich Steimann

Abstract Class Hierarchies, Factories, and Stable Designs

Much of the debate about the general aptness of class hierarchies is rooted in the different objectives taxonomists and implementers are thought to pursue. Designers of conceptual hierarchies tend to embrace Aristotle's principle of *genus et differentiae* leading to a taxonomic hierarchy of categories or types [7], while those with

implementation in mind focus on the reuse of class definitions and polymorphism as made possible by subclassing and inheritance. This has led to an extensive discussion (see [1, 4, 5, 8]) as to whether Square should be a subclass of Rectangle or vice versa, a dilemma that is, of course, precedential in character.

Despite the different perspec-

tives there appears to be a broad consensus that, in principle at least,

- both a conceptual type and a class (as a programming construct) are intensions the extensions of which are sets of instances; and
- the extensions of subtypes are subsets of the extensions of

Copyright of Communications of the ACM is the property of Association for Computing Machinery. The copyright in an individual article may be maintained by the author in certain cases. Content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.