An Application of Rough Set Theory to Modelling and Utilising Data Warehouses

DENG Mingrong[1] YANG Jian-Bo[2] PAN Yunhe[3]

[1] School of Management, Zhejiang University, Hangzhou 310028, P.R.C

[2] Manchester School of Management, University of Manchester Institute of Science and Technology, UK

[3] Zhejiang University, Hangzhou 310027, P.R.C.

Abstract: A data warehouse often accommodates enormous summary information in various granularities and is mainly used to support on-line analytical processing. Ideally all detailed data should be accessible by residing in some legacy systems or on-line transaction processing systems. In many cases, however, data sources in computers are also kinds of summary data due to technological problems or budget limits and also because different aggregation hierarchies may need to be used among various transaction systems. In such circumstances, it is necessary to investigate how to design dimensions, which play a major role in dimensional model for a data warehouse, and how to estimate summary information, which is not stored in the data warehouse. In this paper, the rough set theory is applied to support the dimension design and information estimation.

Keyword: Rough sets, Data warehouse, Dimension;

§ 1. Introduction

Building a data warehouse is an effective way for accommodating enormous historical data that should be subject oriented and organised in hierarchies, and for providing a good environment for knowledge discovery. Dimensional modelling is a logical design technique often used for data warehousing. Every hierarchy related to a concept is reflected in the definition of a dimension.

Summary information at all levels defined using dimensions accounts for most of the contents in a data warehouse, whilst data sources may not be incorporated into the data warehouse because of storage or security reasons or because it is simply not available.

It is often the case that distributed and maybe heterogeneous database systems provide data source for data warehouses. Such systems can be built based upon special operations in a particular organisation. Hence, if source data is rolled up (aggregated values for all levels of hierarchies), different conceptual hierarchies will usually be used. If a data warehouse is set up based on distributed databases, integration is needed.

In (Scotney, 1999), the integration of heterogeneous summary tables is developed. Through restoring summary information to a group of linear equations using the Gauss-Jordan elimination algorithm, the summary information can be integrated and new estimated summary information based on new granularity may be generated. In this paper, we address a different problem. It is assumed that the set of summary data is consistent or has been integrated. It is required to set up a data warehouse to accommodate all summary data without necessarily knowing detailed data. Although many hierarchies can be defined on a dimension, it is necessary to identify how summary data in basic granularity should be defined so that all constraints can be kept consistent. Another problem is how summary information stored in advance could be used to estimate new summary information for an arbitrary set? For example, in an oil refinery, a daily output figure in the report of a refinery unit may be measured at 8:00 every morning, while in the report for inventory the same quantity could be measured at every 4:00 p.m. To the question is how to estimate the output for a period of time, for example, from 6:00 a.m. to 6:00 p.m. of the same day.

It is well recognised that proper identification of granularity is crucial to the usefulness and cost of the warehouse, since granularity at too low a level results in an exponential increase in the size requirements of the warehouse (Humphries et al, 1999). It is not worthwhile to accommodate the summary data for all the possible quests. However, no works for estimating new summary information from the given summary information have been found published.

This paper presents a novel application of rough set theory to address the two problems as described above in data warehousing.

In the rough set theory, exact knowledge can be learnt from relationships among crisp sets (unions of elementary sets). Using the crisp set approximation of any rough sets, more knowledge can be generated.

The paper is organised as following. In section 2, basic approximation concepts in rough set theory are introduced. Their application in data warehousing is described in section 3. Two simple examples are given in section 4. The paper is concluded in section 5.

§ 2.Upper and Lower Approximation in Rough Set Theory

Suppose **U** is a finite set (the universe) of all objects of interest. According to (Pawlak, 1991), a concept in **U** is corresponding to a subset $X \subset \mathbf{U}$, and (abstract) knowledge is corresponding to a family of concepts in **U**. A knowledge base can be defined as a relational system $\mathbf{K}=(\mathbf{U}, \mathbf{R})$, where **R** is a family of equivalence relations over **U**. Obviously for any $\mathbf{P} \subset \mathbf{R}$, $\mathbf{P} \neq \Phi$, $\cap \mathbf{P}$ is also an equivalence relation, which can usually be denoted by IND(**P**). In other words, for any $x, y \in \mathbf{U}$, x IND(**P**) y if and only if, x P y for any $P \in \mathbf{P}$. Define $IND(\mathbf{K})=\{IND(\mathbf{P}): \mathbf{P} \subset \mathbf{R}, \mathbf{P} \neq \Phi\}$. Then for any concept X in **U**, using an equivalence relation $R \in IND(\mathbf{K})$ the following two subsets are called the R-lower and R-upper approximation of X respectively:

$$R_*X = \cup\{Y \in U/R, Y \subseteq X\}$$

$$R^*X = \cup\{Y \in U/R, Y \cap X \neq \Phi\}$$

A set X satisfying $R_*X = R^*X$ is called crisp, with respect to information R. Otherwise it will be called a rough set.

From any set of attributes related to **U**, B, we can define an equivalence relation:

$$IND(B)=\{(x,y) \in \mathbf{U}^2: a(x)=a(y), \forall a \in B\}$$

Equivalently, a knowledge base can be represented using an information system $\mathbf{S}=(\mathbf{U}, \mathbf{A})$, where **A** is a nonempty and finite set of primitive attributes. Every $a \in \mathbf{A}$ defines an information function f_a: $\mathbf{U} \rightarrow V_a$, where V_a is called the domain of a. For any $a \in \mathbf{A}$, $v \in V_a$, (a, v) or in short a_v is a property, which means attribute a has a value of v.

§ 3.Application in Data Warehousing

3.1 Dimensional model of data warehouse

Unlike an operational system, which is often constructed to improve the way computer technology is used in business process, a data warehouse project is typically initiated to satisfy quests for information. Very often, the majority of the contents in a data warehouse are summary data organised in various conceptual hierarchies. It is essential to optimise the structure in which data is stored.

One of the effective ways is to introduce the concept of dimension. Dimensions are parameters over which the analytical processing is performed. One or more hierarchies can be defined on a dimension. By defining a set of dimensions in detail, data related to a particular subject could be represented in a fact table in the middle of a star-like dimensional model.

Using a dimensional model, a database is highly understandable to users, from end users in business to report writers, query tools and program writers, and the model is open to accommodate unexpected new data elements and new design decisions.

In order to realise a dimensional modal most efficiently, many data warehousing products adopt proprietary multidimensional databases. In Oracle Express, the facts — special topic data values based on a set of dimensions are defined by “variables”. For example, “output” can be a variable based on time dimension, unit dimension and product dimension. Once the value of every dimension is given, the data of a variable is determined. Based on hierarchies defined on dimensions, a data-mining user can start with summary data and drill down into detail data by finding regulations or looking for arguments to prove or disprove a hypothesis.

The multidimensional structure is powerful for the users to slice, rotate and analyse data in a visualised way.

3.2 Equivalence relations

It is common that summary data is calculated and stored in multidimensional databases and original data is retrieved from other systems (e.g. relational databases) or is simply inaccessible. In this section, it is assumed that original data is not retrieved.

Suppose original data sources have the simplest form as shown in Fig.1, which will not be included in multidimensional databases.

A practical data warehouse may include many summary variables. In this section, we only consider additive summary variables, such as COUNT, SUM, SUM-OF-SQUARES. Many non-additive summary variables like AVERAGE, INDEX, RATE, etc. can be defined in terms of or derived from additive summary variables (Malvestuto, 1993). Furthermore, since most data in a data warehouse are non-negative or can be represented using a formula, e.g. difference of two non-negative variables, it can be assumed that all data are non-negative.

Roll up hierarchies for any variable can be determined by the hierarchies defined on the related dimensions. A hierarchy on a dimension is depicted as in Fig.2, where the nodes at the bottom level are identical with some of the elements in the original data sources, and each of the nodes at upper levels represents a coarser granularity of this kind of elements.

Elements	Data
E1	D(E1)
E2	D(E2)
.	.
.	.
En	D(En)

Fig.1 the original data source

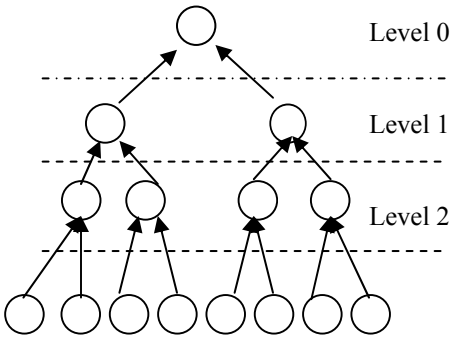

Fig.2 a hierarchy

The question is how summary information can be generated for any given set of elements and the corresponding dimension should be designed if only part of summary information based on a group of rollup hierarchies are available and must be supported by the data warehouse.

For a hierarchy **N** in Fig.2, nodes at level i are defined by N_{i1}, N_{i2}, …, N_{in_i}. Let E={E1, E2,…En}. Then the summary data related to a node N_{ij} is given by,

$$S(N_{ij}) = \sum_{Ei \in N_{ij}} D(Ei)$$

Note that the node set $N_i=\{N_{i1}, N_{i2}, \ldots, N_{in_i}\}$ at every level i defines an equivalence relation on E, $Ep\widetilde{N}_i Eq \Leftrightarrow \exists j \in \{1,..., n_i\}, \{Ep, Eq\} \subset N_{ij}$. Hence, for a hierarchy **N**, there is a set of equivalence relation $\widetilde{N} = \{\widetilde{N}_1, \widetilde{N}_2,..., \widetilde{N}_l\}$ (???), where l is the last level except the level of original elements. For a would-be-data-warehouse, there is a set of hierarchies $\mathbf{N}^1$, $\mathbf{N}^2$,…, $\mathbf{N}^k$ on E, and

$$\widetilde{N}^1 = \{\widetilde{N}^1_1,..., \widetilde{N}^1_{l_1}\},$$
...
$$\widetilde{N}^k = \{\widetilde{N}^k_1,..., \widetilde{N}^k_{l_k}\}.$$

If we denote the family of equivalence relations by $N_{dw} = \widetilde{N}^1 \cup \widetilde{N}^2 \cup ... \cup \widetilde{N}^k$, $K_{dw}=(E, N_{dw})$ defines a knowledge base.

3.3 The rough set approach

The first question is how to define a dimension so that summary information can be kept consistent with the given hierarchies and properly supported by the data warehouse.

One way to answer the question is to define new "original" data sources, which are also summary data with finer granularity than any given hierarchy and can be calculated from the given information. In fact, this is equivalent to define an equivalence relation on E. Obviously, for any elements Ep and Eq, if $Ep\,\widetilde{N}^i_j\,Eq$ for any i=1,…,k, j=1,…,l_i, then Ep and Eq should be in the same equivalence class. Since for any i=1,…,k, $1 \le m < n \le l_i$, we have $\widetilde{N}^i_m \supset \widetilde{N}^i_n$, the coarsest equivalence relation satisfying the condition is $R_e = \widetilde{N}^1_{l_1} \cap \widetilde{N}^2_{l_2} \cap ... \cap \widetilde{N}^k_{l_k}$.

Hence, if summary data are consistent, they can be obtained by rolling up the data along the hierarchies defined on E/R_e, which then can be defined as the set of basic values of the dimension.

The second question is how S(X) for any subset X in E can be estimated based on the model structure with the dimensions defined as above.

Actually, for any equivalence relation R on E, from the non-negativity of the data and the relation $R_*X \subseteq X \subseteq R^*X$, we have $S(R_*X) \le S(X) \le S(R^*X)$. The most accurate estimation for S(X) is given by the following theorem .

Theorem: $S(R_{e*}X)$ and $S(R_e^*X)$ are the accurate lower and upper bounds of S(X), respectively.

Proof: In fact, for the equivalence relation R_e, E/R_e is composed of sets in which any elements are indiscernible with respect to any equivalence relation $R \in N_{dw}$.

Consider the boundary region of X, $R_e^*X - R_{e*}X = \cup\{Y \in U/R_e, Y \cap X \neq \Phi, Y \not\subset X\}$. For every Y in $R_e^*X - R_{e*}X$, $Y=(Y \cap X) \cup (Y \backslash X)$, where $Y \cap X \neq \Phi$, $Y \backslash X \neq \Phi$, and $(Y \cap X) \cap (Y \backslash X) = \Phi$. Hence, S(X) will reach the lower bound $S(R_{e*}X)$ if $S(Y \cap X)=0$ for every Y in $R_e^*X - R_{e*}X$, and the upper bound $S(R_e^*X)$ if $S(Y \backslash X)=0$ for every Y in $R_e^*X - R_{e*}X$. □

Therefore, based on the knowledge base K_{dw}, a procedure for estimating summary information for any set of elements X can be given as follows.

Procedure I

1) i:=0
2) For $R = \widetilde{N}_1^1,...,\widetilde{N}_{l_1}^1,...,\widetilde{N}_1^k,...,\widetilde{N}_{l_k}^k$, if there exists a subset $Y \in E/R$, such that $Y \subset X$, then i:=i+1, X_i:=Y, X:=X\Y.
3) If $X=\Phi$, then $S(X) = \sum_{j=1}^{i} S(X_j)$, STOP.
4) For $R = \widetilde{N}_{l_1}^1 \cap \widetilde{N}_{l_2}^2 \cap ... \cap \widetilde{N}_{l_k}^k$, compute R_*X and R^*X.
5) Estimate $S(R_*X)$ and $S(R^*X)$.

Taking $\{S(Y)|Y \in E/R\}$ as variables $x_1, x_2, \ldots, x_m$, the summary information at $N_{l_1}^1, N_{l_2}^2, ..., N_{l_k}^k$ is corresponding to a group of linear equations:

$$Ax=B.$$

$S(R_*X)$ and $S(R^*X)$ are corresponding to the sums of some members of $x_1, x_2, \ldots, x_m$, namely, for some vector c and C, cx and Cx, respectively.

If equations Ax=B has a unique solution, x_0, then let $m=cx_0$, and $M= Cx_0$.

Otherwise, let m and M be the solutions of the following two linear programmes,

min cx and max Cx

s.t. $Ax=B$ s.t. $Ax=B$

$x \geq 0$ $x \geq 0$

6) The lower bound and upper bound of S(X) are given by $\sum_{j=1}^{i} S(X_j) + m$, $\sum_{j=1}^{i} S(X_j) + M$, respectively.

In some cases, the values of a variable for dimension values cannot be determined uniquely if the matrix A in procedure I does not have full rank. Under such circumstances, more information should be collected if possible. If there is no such information available, the values may be defined by finding a feasible solution of equations Ax=B and storing A and B in other objects, for example tables. Therefore, for any set whose summary information can not be determined uniquely, an estimated interval can be obtained by solving two programmes as shown in step 6 of Procedure I.

§ 4.Illustrative examples

In this section, the rough set approach is demonstrated using two simple examples. The first one adopts the same hierarchies as in (Scotney, 1999). The original elements are described as in the first column of Table 1. The columns 2 and 3 represent the attributes of the corresponding information system, which is obtained by using the two different regional hierarchies used in the two reports (The reports are about summary data of applying the COUNT function to the sales attributes, as shown in Table 2). The attribute A_1 is deduced from the report 1, and A_2 from report 2. For example, in report 1, Cambridgeshire(1), Lancashire(5), Norfolk(6) belong to the same region—Eastern (a node in the hierarchy), so they have the same value for the attribute A_1.

	A_1	A_2
(1) Cambridgeshire	1	1
(2) Derbyshire	3	2
(3) Durham	4	4
(4) Lancashire	5	5
(5) Lincolnshire	1	2
(6) Norfolk	1	1
(7) Northumberland	4	4
(8) Nottinghamshire	2	2

(9) Shropshire	2	3
(10) Warwickshire	2	3
(11) Yorkshire	5	4

Table 1.The counties and the information system

Report 1			**Report 2**		
Region	Counties	Count	Region	Counties	Count
Eastern	1,5,6	285	East Anglia	1,6	221
Midlands	8,9,10	299	E. Midlands	2,5,8	292
Peak Dist.	2	139	W. Midlands	9,10	206
Tyne & wear	3,7	149	North – East	3,7,11	231
Mid-North	4,11	132	North – West	4	50

Table 2.Two reports based on different regional hierarchies

From the information system, we get $E/R_e=E/IND(A_1,A_2)=\{E_1'=\{1,6\}, E_2'=\{2\}, E_3'=\{3,7\}, E_4'=\{4\}, E_5'=\{5\}, E_6'=\{8\}, E_7'=\{9,10\}, E_8'=\{11\}\}$. Hence, a dimension with 8 elements in basic level can be defined, and the two hierarchies can be consistently defined on it. For example, the hierarchy in the report 1 is depicted as in Fig 3.

In order to estimate summary information at any subset X of E, the summary at basic level has to be calculated first. Let $x_1=S\{1,6\}$, $x_2=S\{2\}$, $x_3=S\{3,7\}$, $x_4=S\{4\}$, $x_5=S\{5\}$, $x_6=S\{8\}$, $x_7=S\{9,10\}$, $x_8=S\{11\}$, the known summary data are equivalent to a group of linear equations. For example, from the summary data of "Eastern" in report 1, one can have $x_1+x_5=285$. Thus, the matrix of the constraint equation is easily obtained,

$$A=\begin{pmatrix}1&0&0&0&1&0&0&0\\1&0&0&0&0&0&0&0\\0&0&0&0&0&1&1&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&1\\1&0&0&0&0&0&0&0\\0&1&0&0&1&1&0&0\\0&0&0&0&0&0&1&0\\0&0&1&0&0&0&0&1\\0&0&0&1&0&0&0&0\end{pmatrix}$$

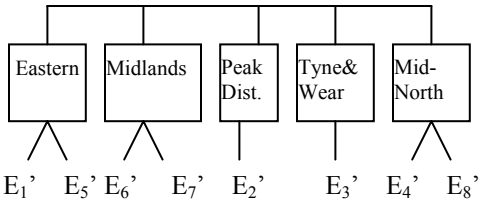

Fig. 3 The new hierarchy in report 1

Since rank(A)=8, we can calculate the values of x_1 to x_8, as follows, x_1=221, x_2=139, x_3=149, x_4=50, x_5=64, x_6=93, x_7=206, x_8=82.

For any given subset X, we can estimate S(X) using $S(R_{e*}X)$ and $S(R_e^*X)$. For example, for X={5,8,9}, $R_{e*}X$={5,8} and R_e^*X={5,8,9,10}. Since S{5,8}=64+93=157, S{5,8,9,10}=64+93+206=363, we get S(X)=157~363.

In some cases, the summary data at the basic level of a dimension can not be determined by the given summary information. Let us consider a simple example, where the original data is based on 5 elements, $\{E_1\}$, $\{E_2\}$, $\{E_3\}$, $\{E_4\}$, $\{E_5\}$. The summary information we know and should be supported by the warehouse is $S\{E_1, E_2\}$, $S\{E_3, E_4, E_5\}$ and $S\{E_1, E_3, E_4\}$, $S\{E_2, E_5\}$.

Obviously, $E/R_e=\{\{E_1\}, \{E_2\}, \{E_3, E_4\}, \{E_5\}\}$. However, $x_1=D(E_1)$, $x_2=D(E_2)$, $x_3=D(E_3)+D(E_4)$, $x_4=D(E_5)$ can not be determined by the information given since the related matrix has not full rank. Hence, the constraint conditions, $x_1 + x_2= b_1$, $x_3 + x_4= b_2$, $x_1 + x_3= c_1$ should be stored. The variable values at the basic level, x_1, x_2, x_3, x_4 can be set to any values that satisfy the conditions.

If we want to estimate the variable value at any subset on the dimension, for example, $X=\{E_1, E_2, E_3\}$. Since $R_{e*}X= \{E_1, E_2\}$, $R_e^*X=\{E_1, E_2, E_3, E_4\}$, we get the lower bound for S(X), b_1, and the upper bound, $b_1+\max\{x_3 | x_1 + x_2= b_1, x_3 + x_4= b_2, x_1 + x_3= c_1, x_1, x_2, x_3, x_4 \geq 0\}$.

§ 5.Conclusion

In this paper, we investigated how to estimate summary information, which is not stored in a data warehouse, and how to design a dimension in the data warehouse when only a sub-set of summary data is given and must be supported in the future.

In the cases where there is inconsistency among summary data, the approach applies to deal with the data after integration.

Reference

1. Oracle Express Database Design and Control, Student Guide, Volume one and two, ORACLE Company, 1997.

2. Oracle Express Server Reviewer's Guide, ORACLE Company, 1998.
3. C. Adamson, M. Venerable, Data Warehouse Design Solutions. John Wiley & Sons, 1998.
4. V.R. Gupta An Introduction to Data Warehousing, http://system-services.com/dwintro.asp
5. M. Humphries, M.W. Hawkins, M.C. Dy, Data Warehousing: Architecture and Implementation, Prentice-Hall, Inc. 1999.
6. W.H. Inmon, J.A. Zachman,J.G. Geiger, Data Stores Data Warehousing and the Zachman Framework, McGraw-Hill Book Co.1997.
7. R. Kimball, The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouse, John Wiley & Sons, 1996.
8. R. Kimball, A dimensional modelling manifesto, http://www.dbmsmag.com/9708d15.html
9. F.M. Malvestuto, A universal-scheme approach to statistical databases containing homogeneous summary tables. ACM Transactions on Database Systems 18:678-708,1993.
10. Z. Pawlak, Rough Sets--Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers,1991
11. B.W. Scotney, S.I. McClean, M.C. Rodgers, Optimal and efficient integration of heterogeneous data. Data and Knowledge Engineering 39:337-350,1999.