An Improved Data Mining Algorithm Based on Concept Lattice

Xiao-feng Li
Department of Computer Science and Technology,
ChengDong College Of Northeast Agricultural
University
Harbin ,150001, china
E-mail:mberse@126.com

Jian-guo Luo, An Shi
Information Center of National Defense Science and
Industry Bureau
Beijing, 100000,China
E-mail:cidexam@163.com

***Abstract*—In order to solve the multidimensional data model and relational data model, query between the two-way data system query, data cleansing, data conversion, centralized, distributed data accuracy and consistency control problem, this paper described the concept of grid-related, the global data mining combined with local data mining, is proposed based on local information based on the concept of a global grid of data mining algorithm, and the mining process was divided into ETL action, combined with the ETL process workflow, vast amounts of data distributed parallel sequence mining. Experiments show that the algorithm has a good effect on enhanced data processing capability.**

Keywords-component; data; conceptlattice; datamining; integration; Timing mining

I. INTRODUCTION

Concept lattice[1], also called Galois lattice, has received increasing concerns for being taken as formal concept analysis and expression tool since Formal Concept Analysis (FCA) was proposed by Wille in the 1980s [2]. Lots of important achievements have been made too. The definition of concept lattice which is generally recognized for the time being is that it is a kind of effective tool for knowledge expression and processing. The basic idea is a hierarchical structure formed on the basis of binary relation, an effective instrument for data analysis and rules extraction.

Research findings indicate that there have been so far many algorithms used to construct concept lattice. The application of the technique of concept lattice is relatively mature in foreign countries, focusing on the study of rough set, fuzzy set, ontology and semantic Web. Consulting documents to discuss concept lattice is made by the two classes of methods [3]: one is incremental algorithm, which generates concept set or Hasse diagram of the i^{th} object in the formal context; the other is batch algorithm, which builds concept lattice and Hasse diagram according to the whole formal context. [4] elaborates all research methods for formal concept analysis both home and abroad. Studies on such fundamental theories as rules mining, attribute reduction, sublattice and quotient lattice, building algorithm of concept lattice are the focus of incremental algorithm. But, very few are seen with regards to batch algorithm for massive data due to limitations by the environment or natural conditions [5].

In the paper, it introduces an improved data mining method of the global concept lattice which relies on local information after related studies on concept lattice and through the integration of global and local data mining. The mining process is decomposed into ETL actions, i.e. combining time series data mining technology with ETL to eventually realize time series mining of parallel distributed mass data with the help of ETL processing workflow. Yet, we use the new method for the information integration in manufacturing industry. Empirical analysis shows its effectiveness.

II. RELEVANT WORK

A. Concept lattice

Concept lattice is a type of induced lattice which comes into being based on the partial ordering relation between R and O, D, as well as the binary relation R between example set O and attribute set D in the triple collection T=(O, D, R) in a fixed formal context.

Lattice construction is actually the clustering of concepts. In the concept lattice, lattice construction algorithm plays a critical role. For the same batch of data, the generated lattice is only, not influenced by data or attribute sequence. It is one of its advantages.

B. ETL workflow

ETL (Extract-Transform-Load) workflow is defined here as business rules for data processing and transformation, which includes data extraction, cleaning, conversion and uploading, to define data processing rules and tasks based on the workflow and also through it to control the implementation of various data processing tasks.

For the integration of the workflow technology and concept lattice, it is essential to realize real-time incremental ETL process. A great number of scholars have studied real-time incremental ETL technology and presented the three methods by using timestamp, trigger and logs [6]. As far as the technical implementation is concerned, the above three methods are mature. The key to real-time incremental ETL is that data in the data warehouse are historical and stable, while data in business system are changeable from time to time, that is, one business data would be added, corrected, deleted or restored within a period. So, real-time extraction and uploading would easily cause inconsistency between the summarized data in the warehouse and current business data. That is the key issue we'll discuss in the following parts.

Published by Atlantis Press, Paris, France.

III. IMPROVEMENT OF DATA MINING ALGORITHM BASED ON CONCEPT LATTICE

Data mining algorithms which are generally based on concept lattice simply use data mining technologies to perform a series of data discoveries. Here it combines time sequence data mining techniques and ETL to investigate them by taking instance of time series trend mining based on multiple regression analysis. Sample data acquisition, datacompilation,datapre-processing,complicatecomputation, partial correlation, parametric estimation, variable selection, model checking and the like are designed as ETL's actions. Then with the advantage of workflow technology, the execution order and parameter passing of those actions are reasonably organized.

A. Initialization

1)Determine dependent variable Y and independent X1, X2, …, Xm and the number m of independent variables of regression analysis, as well as level of significance; define samples n used for building multiple linear regression model (when n is unknown, select all original data); define the accuracy of computation (the length of decimal places of floating point numbers are retained during computation); specify the data source for regression analysis, from which variables originate. See Table 1.

2)Set up the temporal data table temp (head: item no., X1, X2, …, Xm, Y), as indicated in Table 1.

3)Acquire the original data and input values of X1, X2, …, Xm and Y to the temp.

B. Partial Correlation

Calculate successively the partial correlation coefficients in the $(m-1)^{th}$ order between dependent variable Y and each independent variable Xi (1≤i≤m) according to the data in temp, and then put them into one-dimensional array rp[m] as:

Define the array m [] and save the sequence ki=1,2…(i-1)(i+1)…m in Rki=1,2,…(i-1)(i+1)…m to indicate the availability. Reckon Rki=1,2,…(i-1)(i+1)…m and directly call Rp= PR(0，i，m[]);

- Function PR（k，i，m[]）
  Since values of samples of both Y and Xi (i=1,2, …, m) are all known, we can use PR (k，i，m[]) to obtain the partial correlation coefficients between Y and each Xi.
- Function R (k,i)
  A new array flag[m.length-1] should be established to make flag[j]=m[j] as regression would change the marker bit of m, where, 0≤j≤m.length-1. Next, use R (k,i) to get the simple correlation coefficient between Xk and Xi and return the value.

C. Variable Selection

1)According to the level of significance a, check the data table and get Fmin of F which can introduce variables into regression equation;

2)Build dynamic arrays Xj and Xs, then write down independent variables which have been imported into regression equation and those not been into the equation. The initial value of Xj is 0, Xs' is {X1，X2，…，Xm};

3)Invoke simple linear regression analysis programme to analyze variable association of each independent of Xs and each dependent of Y, then, ti which is verified by regression equation t can be concluded and is placed into ta;

4)Values in the dynamic array T being known, it is likely to compute the maximum value introduced by F among each independent variable of Xi and the corresponding Xi, which are then put into Fmax and X;

5)If Fmax is less than Fmin or Fmax less than ta, then exits; or, record the regression equation where X belongs to, testing value and that of ta; also, add Xi which is in X to Xj while delete Xi from Xs;

6)Analyze according to data in temp sheet each independent variable of Xs and all independents of Xj and all dependents of Y, as to get each regression equation before acquiring values of new independent Xi (Xi is the element of Xs) and ta which is inspected by t. Such values are put in the dynamic array T. Obtain the inspection value F of significance of each equation and put them in the dynamic array F; Fa/2 is available through table look-up according to the level of significance and is put in Fa;

7)Use data in the dynamic array T to reckon the biggest imported value by F and relevant Xi of each newly independent variable Xi, and put those values in Fmax and X;

8)If Fmax is less than Fmin, exits; or, record the regression equation as well as values of F and Fa/2 (put in Fa) of the independent variable of the biggest F-imported value (for the dynamic array F, F[j] accords to the F-tested value of the equation where xi belongs);

9)If F[j]<Fa, exits; otherwise, add Xi which is put in X to Xj while remove Xi from Xs;

10)Repeat the above operations till Xs is empty.

D. Trend Analysis Model

1)Modify temp sheet by adding X0 to the "Item NO." field and that we get the observed value is all 1;

2)Calculate XTX and put it in two-dimensional array XtX[m+1][m+1]; then again calculate XTX and put it in one-dimensional array XtY[m+1], which is shown like:

- Compute the upper triangular including the leading diagonal and XTY of matrix XTX;

```
for(i=0; i《=m;i++)
{
for(j=i; j《=m;j++)
Accumulate the product of the observed values in both Xi and Xj fields in the temp sheet;
Accumulate the product of the observed values in both Xi and Y fields in the temp sheet;
}
```

where, tempi and k mean the k^{th} observed value in Xi field; tempj and k mean the k^{th} observed value in Xj field;

- Use matrix XTX to achieve the symmetry of principal diagonals and fill in the lower triangular part (exclusive of principal diagonals). The algorithm shows:

```
    for(i=0; i《=m;i++)
    for(j=i; j 《=m;j++)
    XtX[j][i] = XtX[i][j];
```

- Count in proper order the statistical magnitude t of the i^{th} ($0\leq i\leq m$) independent variable, then put it in the one-dimensional array T[i-1];
- Look up t distribution table to get ta/2 and put it in ta;
- Calculate the statistical magnitude F with the use of data in temp table and put it in F;
- Look up F distribution table to get Fa/2 and put it in Fa;
- Output: F, Fa

IV. Empirical Analysis

Based on the above arithmetic design, we constructed a data integration system in one auto manufacturing enterprise and carried it out for applications, which is based on the ETL workflow. The system consists of data processing server, processing rules repository as well as data processing rules development and design tools. Of them, data processing server includes ETL Server and Meta Server.

Repository saves all meta data during ETL process, i.e. information bank in the data warehouse system, which contains meta data of other related products. Meta data in the repository can be available in the form of XML papers with embedded data base in no need of the support of DBMS from the third party. Besides, they can provide means for backup and recovery. Such meta data have the following contents:

1)Definition of data source: referring to connection parameters in relational database, storage location URI of XML papers;

2)Target database: definition of the location of target database;

3)Database structure definition: including the structural description of both data source and target database;

4)ETL logic rules: business rules with regards to data extraction, cleaning, conversion and uploading; such rules depend on the data structure of both data source and target database, nothing to do with specified data source and target database;

5)ETL operating rules: the actual operational plan of ETL logic rules, composed of various meta data such as ETL logic rules, data source, target database, execution cycle, runtime environment.

6)ETL execution log: contains all information produced during every one of ETL rules running process.

To validate the effectiveness of the proposed method, we compared the efficiency by manual handling of ETL actions before the implementation and that by the system.

We made the test on a PC, 2GB memory, P4 3.0GHz processor, with Windows XP operating system. Experimental data are two groups of sample data which are randomly generated. The number of attributes averagely owned by objects in the formal context is ‖D‖=40, 60; the number of objects is ‖T‖=100000, 200000,…, 800000, which produce randomly two groups of testing data sets.

In order to reflect the precision of the mentioned method in a more intuitive way, we distributed evenly the number of objects into n (n=2, 4, 8) sub data sets. Figure 1shows the accuracy after being treated by manual method. Figure 2 shows the accuracy after being processed by the algorithm. It can be observed from figure 1 that in the case of more abundant data, objects are in direct proportion to the accuracy, which verifies how effective the new method is for processing enormous data.

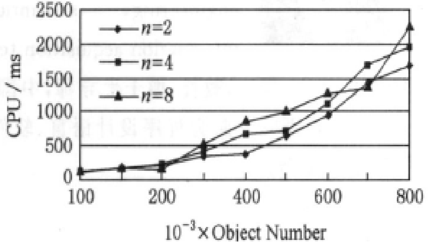

Figure. 1 Accuracy by Manual Method

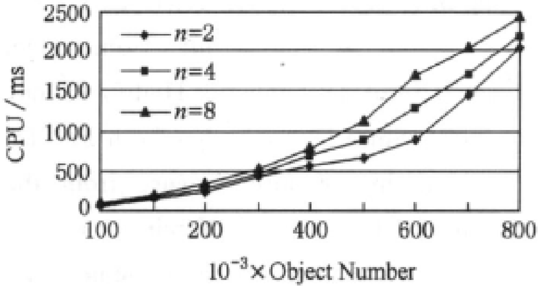

Figure.2 Accuracy by the Proposed Method

V. Using the Template

Through the combination of global and local data mining and based on the global data mining of local information, the algorithm's efficiency is improved with the employment of local information; on the other hand, more understandable models can be obtained through the integration of local information. What's innovative to the method discussed here is it decomposes data mining process into different ETL actions, to be specific, combining time series data mining technology and ETL together with ETL workflow to make successfully the temporal discovery of massive data in parallel distribution.

References

[1] Yao Y Y. A comparative study of formal concept analysis and rough set theory in data analysis. Proc of the Rough Sets and Current Trends in Computing, LNCS 3066. Berlin:Springer, 2004,pp. 59-68

[2] Zhang Kai, Hu Yunfa, Wang Yu. An IRST-based algorithm for construction of concept lattices. Journal of Computer Research and Development, 2004, pp. 1493-1499

[3] Li H R, Zhang W X, Wang H. Classification and reductionof attributes in concept lattices.Proc of IEEE Int l Conf on Granular Computing. Los Alamitos: IEEEComputer Society, 2006, pp. 142-147

[4] Wei Ling. Reduction theory and approach to rough set andconcept lattice. Xi an: Xi an Jiaotong University, 2005

[5] Zhang Wenxiu, Wei Ling,et al. Attribute reduction inconcept lattice based on discernibility matrix Proc of the 10th Int l Conf on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing (RSFDGrC), LNCS 3642. Berlin:Springer, 2005, pp. 157-165

[6] Shao M W. The reduction for two kind of generalized concept lattice .Proc of the 4th Int l Conf on Machine Learning and Cybernetics. Berlin: Springer, 2005, pp. 2217-2