IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4914

Knowledge Discovery in Data Warehouse Environment

M. Mehdi Owrang O.
Department of Computer Science, Audio Technology, and Physics, Ameircan University, Washington, DC 20016,
owrang@american.edu

ABSTRACT

Current database technology involves processing a large volume of data in order to discover new knowledge. However, knowledge discovery on just the most detailed and recent data does not reveal the long-term trends. A data warehouse is an ideal environment for knowledge discovery since it contains the cleaned, integrated, detailed (most recent operational), summarized, historical, and meta data. A key issue in any discovery system is to ensure the consistency, accuracy, and completeness of the discovered knowledge. We discuss the benefits and issues in knowledge discovery in data warehouse.

INTRODUCTION

Modern database technology involves processing a large volume of data in databases in order to discover new knowledge. Knowledge discovery is defined as the nontrivial extraction of implicit, previously unknown, and potentially useful information from data [1, 2, 5, 6, 8].

The automatic knowledge acquisition in a non-data warehouse environment has been on the operational databases which contain the most recent data about the organizations. Summary and historical data, which are essential for accurate and complete knowledge discovery, are generally absent in the operational databases. Rule discovery based on just the detailed (most recent) data is neither accurate nor complete. In addition, the design of an operational relational database is based on the normalization technique which is not suitable for effective knowledge discovery. A data warehouse is an ideal environment for rule discovery since it contains the cleaned, integrated, detailed, summarized, historical, and meta data [4, 8, 11, 13].

In the following sections, we explain how/why a data warehouse can provide an effective environment for discovering accurate, complete, consistent, and meaningful rules. We look at the knowledge discovery process on detailed, summary, and historical data. Also, we show how the discovered knowledge from these data sources can complement and validate each other.

DATA WAREHOUSES

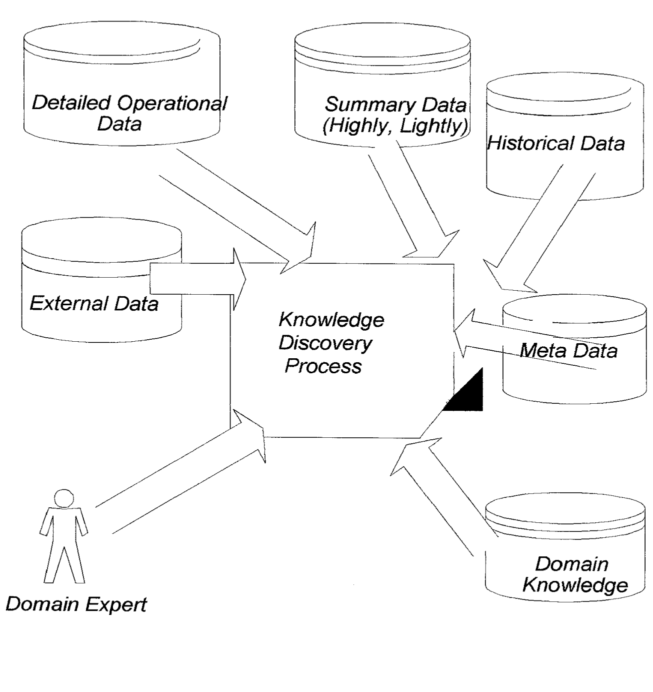
Most of the knowledge discovery has been done on operational relational databases. However, such knowledge discovery in operational environment could lead to inaccurate and incomplete discovered knowledge. Without first warehousing its data, an organization has lots of information that is not integrated and has little summary or history information. The effectiveness of knowledge discovery on such data is limited. A data warehouse environment integrates data from variety of source databases into a target database that is optimally designed for decision support. A data warehouse includes [1, 3, 4, 11] integrated data, detailed and summary data, historical data, and meta data. Each of these elements enhances the knowledge discovery process:

There are several benefits in rule discovery in a data warehouse environment.

1. In a data warehouse environment, the validation of the data is done in a more rigorous and systematic manner. Using meta data, many data redundancies from different application areas are identified and removed. In addition, a data cleansing process is used in order to create an efficient data warehouse by removing certain aspects of operational data, such as low-level transaction information, which slow down the query times [1, 3, 4, 11]. The cleansing process will remove duplication and reconcile differ ences between various styles of data collection.
2. Operational relational databases, built for on-line transaction processing, are generally regarded as unsuitable for rule discovery since they are designed for maximizing transaction capacity and typically having a lot of tables in order not to lock out users. Also, they are normalized to avoid update anomalies. Data warehouses, on the other hand, are not concerned with the update anomalies since update of data is not done. This means that at the physical level of design, we can take liberties to optimize the access of data, particularly in dealing with the issues of normalization and physical denormalization. Universal relations can be built in the data warehouse environment for the purposes of rule discovery, which could minimize the chance of undetecting hidden patterns.

Figure 1 shows a general framework for knowledge discovery in a data warehouse environment. External data, Domain knowledge (data that is not explicitly stored in the database; i.e., male patient can not be pregnant), and Domain expert are other essential components to be added in order to provide an effective knowledge discovery process in a data warehouse environment.

Figure 1: A Framework for Knowledge Discovery in Data Warehouse Environment


Detailed Operational Data
Summary Data (Highly, Lightly)
Historical Data
External Data
Knowledge Discovery Process
Meta Data
Domain Expert
Domain Knowledge

KNOWLEDGE DISCOVERY IN DATA WAREHOUSES

Knowledge Discovery from Detailed Data

Most of the knowledge discovery has been done on the operational relational databases. An operational database stores the most recent and detailed data. In addition, the goal of the relational databases is to provide a platform for querying data about uniquely identified objects. However, such uniqueness constraints are not desirable in knowledge discovery environment. In fact, they are harmful since, from data mining point of view, we are interested in the frequency with which objects occur [1]. In the following, we discuss two main problems associated with the knowledge discovery in the operational relational databases; namely, the possibility of discovering incorrect and incomplete knowledge.

Incorrect knowledge discovery from relational databases

In general, summary data (aggregation) is never found in the operational environment. Without discovery process on summary data, we may discover incorrect knowledge from detailed operational data. Discovering rules based just on current detail data may not depict the actual trends on data. The problem is that statistical significance is usually used in determining the interestingness of a pattern [7]. Statistical significance alone is often insufficient to determine a pattern's degree of interest. A "5 percent increase in sales of product X in the Western region", for example, could be more interesting than a "50 percent increase of product X in the Eastern region". In the former case, it could be that the Western region has a larger sales volume than the Eastern region, and thus its increase translate into greater income growth.

The following example [10] shows that we could discover incorrect knowledge if we only look at the detailed data. Consider the Table A1, where the goal of discovery is to see if product color or store size has any effect on the profits. Although the data is not large, but it shows the points.

Assume we are looking for patterns that tell us when profits are positive or negative. We should be careful when we process this table using discovery methods such as simple rules or decision trees. These methods are based on probabilities that make them inadequate for dealing with influence within aggregation (summary data). A discovery scheme based on probability may discover the following rules from Table A1:

Rule 1: IF Product Color=Blue Then Profitable=No CF=75%
Rule 2: IF Product Color=Blue and Store Size> 5000 Then Profitable=Yes CF=100%

The results indicate that blue products in larger stores are profitable; however, they do not tell us the amounts of the profits which can go one way or another. Now, consider the modified table, where the third row in Table A1 is changed for the Profit to be 100 instead of 7000. Rules 1 and 2 are also true in the modified table. That is, from a probability point of view, Tables A1 and the modified one produce the same results.

However, this is not true when we look at the summary Tables (product color=Blue Profit=6400, based on Table A1) and (product color=Blue Profit=-500, based on modified A1 table). The former summary table tells us that Blue color product is profitable and the latter summary table tells us it is not. That is, in the summary tables, the probability behavior of these detailed tables begins to diverge and thus produce different results. We should be careful when we analyze the summary tables since we may get conflicting results when the discovered patterns from the summary tables are compared with the discovered patterns from detailed tables. In general, the probabilities are not enough when discovering knowledge from detailed data. We need summary data as well.

Table A1: Sample Sales Data

Product	Product Color	Product Price	Store	Store Size	Profit
Jacket	Blue	200	S1	1000	-200
Jacket	Blue	200	S2	5000	-100
Jacket	Blue	200	S3	9000	7000
Hat	Green	70	S1	1000	300
Hat	Green	70	S2	5000	-1000
Hat	Green	70	S3	9000	-100
Glove	Green	50	S1	1000	2000
Glove	Blue	50	S2	9000	-300
Glove	Green	50	S3	5000	-300
Glove	Green	50	S3	9000	-200

Incomplete knowledge discovery from relational databases

The traditional database design method is based on the notions of functional dependencies and lossless decomposition of relations into third normal forms. However, this decomposition of relation is not useful with respect to knowledge discovery because it hides dependencies among attributes that might be of some interest. To provide maximum guarantee that potentially interesting statistical dependencies are preserved, knowledge discovery process should use the universal relation [12] as opposed to normalized relations in order to reveals all the interesting patterns.

Consider the relations Sales (Client Number, Zip Code, Product Purchased) and Region (Zip Code, City, Average House Price) [1] which are in third normal form. The relation Sales-Region (Client Number, Zip Code, City, Average House Price, Product Purchased) shows the universal relation which is the join of the two tables Sales and Region. From the universal relation Sales-Region, we may discover that there is a relationship between the Average Price of the House and the type of Products Purchased by people. Such relationship is not that obvious on the normalized relations.

One possible scheme for validating the completeness/incompleteness of the discovered knowledge is to analyze the discovered rules (known as statistical dependencies) with the available functional dependencies (known as domain knowledge). If new dependencies are generated that are not in the set of discovered rules, then we have an incomplete knowledge discovery. For example, processing the Sales relation, we may discover that if Zip Code=11111 then Product Purchased = Wine with some confidence. We call this a statistical dependency that indicates that there is a correlation (with some confidence) between the Zip Code and the Product Purchased by people. Now, consider the Region relation, where the given dependencies are Zip Code —> City and City —> Average House Price which gives the derived new functional dependency Zip Code —> Average House Price due to the transitive dependency. By looking at the discovered statistical dependency and the new derived (or a given dependency in general), one may deduce that there is a relationship between the Average House Price and the Product Purchased (with some confidence). If our discovery process does not generate such a relationship, then we have an incomplete knowledge discovery that is the consequence of working on normalized relations as opposed to universal relation.

Knowledge Discovery from Summary Data

In knowledge discovery, it is critical to use summary tables to discover patterns that could not be, otherwise discovered from operational detailed databases. Knowledge discovery on detailed data is based on statistical significance (uses probability), which may not detect all patterns, or may produce incorrect results as we noted in the previous section. Summary tables have hidden patterns that can be discovered. For example, a summary table (Product Color, Profit) , based on table A1, tells us that Blue products are profitable. Likewise, a summary table (Product, Profit), based on table A1, tells us that Hat products are not profitable. Such discovered patterns can complement the discoveries from the detailed data (as part of the validation of the discovered knowledge, explained below).

Accurate knowledge, however, cannot be discovered just by processing the summary tables. The problem is that, the summarization of the same data set with two summarization methods may produce the same or different results. Therefore, it is extremely important that the users be able to access meta data that tells them exactly how each type of summarized data was derived, so that they understand which dimensions have been summarized and to what level. Otherwise, we may discover inaccurate patterns from different summarized tables. For example, based on summary tables from table A1, it is the Green Hat in small stores (Store Size <=1000) that make profit and that it is the Green Hat product in large stores (Store Size > 1000) that lose money. This fact can only be discovered by looking at all different summary tables and knowing how they are created (i.e., using the meta data).

Validating possible incorrect rules

It is possible to use the patterns discovered from the summary tables to validate the discovered knowledge from the detailed tables. The following cases are identified for validating possible incorrect/correct discovered rules.

CASE 1: If the discovered pattern from the summary tables completely supports the discovered knowledge from the detailed tables, then we have more confidence on the accuracy of the discovered knowledge.

CASE 2: The patterns discovered from the detailed and summary tables support each other, but they have different confidence factors. Since the discovered patterns on the summary tables are based on the actual values, they represent more reliable information compared to the discovered patterns from the detailed tables which are based on the occurrences of the records. In such cases, we can not say that the discovered pattern is incorrect, but rather it is not detailed enough to be considered as an interesting pattern. Perhaps, the hypothesis for discovering the pattern has to be expanded to include other attributes (i.e., Product or Store Size or both) in addition to the Product Color.

CASE 3: The patterns discovered from the detailed and summary tables contradict each other. The explanation is the same as the one provided for case 2.

CASE 4: There are cases, where the discovered knowledge from summary tables is based on statistical significance. If the discovered knowledge from detailed and summary tables support each other with different confidence factor, then additional information from other sources (perhaps from domain expert, if possible) is need to verify the accuracy of the discovered knowledge.

Knowledge Discovery from Historical Data

Knowledge discovery from operational/detailed or summary data alone may not reveal trends and long-term patterns in data. Historical data should be an essential part of any discovery system in order to discover patterns that are correct over data gathered for a number of years as well as the current data. For example, we may discover from current data a pattern indicating an increased in the students' enrollment in the universities in the Washington DC area (perhaps due to good Economy). Such pattern may not be true when we look at the last five years data.

Using historical data for knowledge discovery

There are several schemes that could be identified in using historical data in order to detect undiscovered patterns from detailed and summary data, and to validate the consistency/accuracy/completeness of the discovered patterns from the detailed/summary data.

1) Validate discovered knowledge from detailed/summary data against historical data

We can apply the discovered rules from detailed and/or summary data to the historical data to see if they hold. If the rules are strong enough, they should hold on the historical data. A discovered rule is inconsistent with the database if examples exist in the database that satisfy the condition part of the rule, but not the conclusion part [7]. A knowledge base (i.e., set of discovered rules from detailed and summary data) is inconsistent with the database if there is an inconsistent rule in the knowledge base. A knowledge base is incomplete with respect to the database if examples exist in the database that do not satisfy the condition part of any consistent rule.

If there are inconsistent rules, then it means we have some historical data that contradict the rules discovered from detailed/summary data. It means we may have anomalies in some of the historical data. This is the case where any knowledge from external data, domain expert, and/or domain knowledge could be used to verify the inconsistencies. Similarly, if we have incomplete knowledge base, then there are some historical data that could represent new patterns or some anomalies. Again, additional information (i.e., domain expert) is necessary to verify that.

2) Compare the rules discovered from detailed/summary data with the ones from historical data

We perform the knowledge discovery on the historical data and compare the rules discovered from the historical data (call it H_RuleSet) with the ones discovered from detailed/summary data (call it DS_RuleSet). There are several possibilities as follows:

A) If $H\_RuleSet \cap DS\_RuleSet = \varnothing$ Then, none of the rules discovered from detailed/summary data hold on the historical data.

B) If $H\_RuleSet \cap DS\_RuleSet = X$ Then

- If $DS\_RuleSet - X = \varnothing$ Then, all of the rules discovered from detailed/summary data hold on the historical data.
- If $X \subset DS\_RuleSet$ Then , There are some rules discovered from detailed/summary data that do not hold on the historical data (i.e, $N\_RuleSet - X$). We can find the data in the historical data that do not support the rules discovered from the detailed/summary data by finding the data that support the rules in N-RuleSet and subtract it from the entire historical data. This data can then be analyzed for anomalies.

C) If $H\_RuleSet - DS\_RuleSet \mathrel{!=} \varnothing$ (or $DS\_RuleSet \subset X$) Then, there are some rules discovered from historical data that are not in the set of rules discovered from the detailed/summary data. This means we discovered some new patterns.

CONCLUSION AND FUTURE DIRECTION

Data warehouses provide a better environment (compared to the operational / transactional environment) for knowledge discovery. However, there are several issues/concerns that need to be addressed before we could have an effective knowledge discovery process. The followings are some of the main issues:

1. The larger a warehouse, the richer its patterns would be. However, after a point, if we analyze "too large" a portion of a warehouse, patterns from different data segments begin to dilute each other and the number of useful patterns begins to decrease [13]. We could select segment(s) (i.e., a particular medication for a disease), to data that fits a particular discovery objective. Alternatively, data sampling can be used to faster data analysis. However, we lose information because we throw away data not knowing what we keep and what we ignore. Summarization may be used to reduce data sizes; although, it can cause problem too, as we noted.
2. Traditionally, most of the data in a warehouse has come from internal operational systems such as order entry, inventory, or human resource data. However, external sources (i.e., demographic, economic, Point-Of-Sale, market feeds, internet) are becoming more and more prevalent and will soon be providing more content to the data warehouse than the internal sources. The next question is then, how do we process these external sources efficiently to retrieve relevant information and discover new knowledge that could explain the behavior of the internal data.

3. Most of the available knowledge discovery tools (i.e., IDIS, KnowledgeSeeker) operate on a single relation or table. If the relevant data are spread over several relations, join operations should be performed on these relations to collect relevant data before the discovery tool is applied. In many cases, the separate relations of a relational database can be logically joined by constructing a Universal Relation (UR) [9]. A UR is either computed and stored, or, if too large, logically represented through a UR interface. A discovery tool should be able to interact with the UR interface and treat the database as a single, flat file (though perhaps inefficient).

REFERENCES

1. Adriaans, Pieter and Dolf Zantinge, Data Mining, Addison-Wesley , 1996.
2. Agrawal, Rakesh; Tomasz Imielinski; and Arun Swami, “Database Mining: A Performance Perspective”, IEEE Transactions on Knowledge and Data Engineering,Vol.5,No.6,Dec. 1993,PP. 914-925.
3. Barquin, Ramon and Herbert A. Edelstein, Building, Using, and Managing The Data Warehouse, (Prentice-Hall PTR, New Jersey, 1997).
4. Bischoff, Joce and Ted Alexander, Data Warehouse-Practical Advise From the Expert, (Prentice-Hall, New Jersey, 1997).
5. Brachman, Ronald J.; Tom Khabaza; Willi Kloesgen; Gregory Piatetsky-Shapiro; and Evangelos Simoudis, Mining Business Databases, CACM, Vol.39, (1996) 42-28.
6. Fayyad, Usama, Data Mining and Knowledge Discovery: Making Sense out of Data ,IEEE Expert, Vol. 11 (1996)20-25.
7. Giarrantanto, Joseph and Gary Riley, Expert Systems - Principles and Programming,(PWS - Kent Publishing Company, Mass.,1989).
8. Inmon, W.H., The Data Warehouse and Data Mining, CACM, Vol. 39 (1996) 49-50.
9. Maier, David, The Theory of Relational Databases, Computer Science Press, 1983.
10. Matheus, Christopher J.; Philip K. Chan; and Gregory Piatetsky-Shapiro,” Systems for Knowledge Discovery in Databases”, IEEE Transactions on Knowledge and Data Engineering, Vol. 5, No. 6, Dec. 1993, PP. 903-913.
11. Meredith, Mary Edie and Aslam Khader, Designing Large Warehouses, Database Programming & Design, (1996) 26-30.
12. Parsaye, Kamran; Mark Chignell; Setrag Khoshafian; and Harry Wong, “Intelligent Data Base and Automatic Discovery”, Neural and Intelligent Systems Integration, by Branko Soucek and the IRIS Group, John Wiley & Sons, Inc, 1991.
13. Parsaye, Kamran, Data Mines for Data Warehouses, Supplement to Database Programming & Design, Vol. 9 (1996). S6-S11.