How To Solve It By Knowledge Mining

Pedro Rafael Falcone Sampaio
e-mail : prafael@lia.ufc.br prafael@uece.br
Labóratorio de Inteligência Artificial (LIA)- Univ. Federal do Ceará
Departamento de Computaçao - Campus do PICI
CEP : 60455-760 Fortaleza - CE Brasil
Fone: 55(85)2439692 55(85)2248066 FAX: 55(85)2231333

Abstract

Frequently we become amazed with the increasing number of problems to be solved that flourish while facing daily activities. Often, related to these problems we have also an incredible amount of data. Since we cannot allways afford time and resources to solve them, we keep on gathering and storing data in large databases, widening the gap between raw and interpreted data. At this point we should reflect about Polya's *maxima* "A great discovery solves a great problem" and realize that databases encompass the knowledge necessary for guiding the decision making process. The question that remains is how to organize and explore this knowledge. This paper presents some approaches to knowledge discovery in databases found in the literature, analyzing issues in classifying and clustering large data sets.

1 Introduction

Next-generation database applications will deal with the explosive growth in the quantity of data stored and the requirements of running complex ad-hoc queries in an attempt to discover possible regularities from data [21], culminating with the need for techniques and tools for understanding and extracting knowledge enclosed in the database [15].

Knowledge Discovery is the nontrivial extraction of implicit, previously unknown and potentially useful information from data [10]. In some applications, examples are Satellite Image Processing, Census and Genetic Databases, present technology allows the gather of massive amounts of data, but yet has not provided means to interpret it at a similar rate. This widens the gap between raw and interpreted data. In order to cope with these requirements, Artificial Intelligence methods are being assessed to support large-scale tasks of data analysis.

Traditionally, the Artificial Intelligence community has contributed with several important technological advances in the database area. Examples of the success of such interdisciplinary applications are the fields of Deductive Databases [22], Active Databases [19] and Knowledge Base Management Systems [3]. A recent contribution of Artificial Intelligence is the usage of Machine Learning strategies to perform knowledge mining in databases.

The two fundamental learning strategies employed are:

- **Learning from Examples**, where given a set of examples and counterexamples of a concept, the learner induces a general concept description that describes all of the positive examples and none of the counter[8]. This strategy is adopted in [1].

- **Learning by Observation**, where given a set of observations (facts), the system acquires concepts that organize those observations and use them in classifying future experiences. This type of concept formation can occur in the absence of a tutor and it can take place even in the presence of irrelevant and incomplete information[11]. It requires a greater amount of inference than the previous strategy and faces the problem of deciding how to manage the available time and resources in acquiring several concepts at once. [9, 2, 4, 14, 13] follow this approach.

Both concept aquisition (Learning From Examples) and concept formation without teacher (Learning by Observation) are instances of the inductive learning paradigm. An inductive learning system generates knowledge by drawing inductive inferences from the given facts under the guidance of *background knowledge*. The background knowledge contains previously learned concepts, goals, bias and inference rules [17].

A formulation of the general paradigm of inductive inference is:

$$DiscoveredRules \wedge BackgroundKnowledge \Rightarrow Facts$$

This shows that the discovered rules are hypotheses that logically imply the facts stored in the database. A Bias for the preferred rules must be given, guiding the discovery process. The reliability of the discovered rules must be evaluated by the recipients of the discoveries. Normally, certainty measures and support constraints (number of instances in the database that participate in the elaboration of the rule) are established to select relevant rules.

A related issue is the computational complexity of the learning process. In Valiant's work [23], it is shown the feasibility of designing learning systems with the following properties:

- The system can learn classes of concepts.
- The classes reflect general-purpose knowledge.
- The computational task of deducing the rules requires polynomial time.

According to Valiant, a learning machine consists of a *Learning Protocol* and a *Deductive Procedure*, both related to a knowledge representation scheme. The former specifies the manner in which information [1] is obtained from the outside. The latter is the mechanism by which a correct recognition algorithm or rule for the concept to be learned is deduced.

There are specific classes of concepts that are learnable in polynomial time using learning protocols as those described in [23]:

1. Conjunctive Normal Form Expressions with bounded number of literals in each clause
2. Monotone Disjunctive Normal Form expressions
3. Arbitrary Expressions in which each variable occurs just once.

[1] Data and Possibly positive examples.

This approach differs from most part of the Learning Strategies which use induction as the mechanism by which concepts are learned [18]. In inductive Learning Strategies, the complexity of Learning Processes achieves exponential rates, requiring background knowledge and heuristics to allow acceptable run-time performance.

The patterns discovered by the learning system may be represented as logical formulas, decision trees, formal grammars, production rules, frames, graphs or even by relational tables[9]. From a logical perspective, database relations can be viewed as disjunctions of conjunctions of literals:

Stock No.	*Description*	*Wholesale*
24	Disk Drive	55
32	Monitor	89
48	Keyboard	77

$$(24 \wedge DiskDrive \wedge 55) \vee (32 \wedge Monitor \wedge 89) \vee (48 \wedge Keyboard \wedge 77)$$

This paper addresses the problem of mining in databases. The scope is restricted to issues in classifying and clustering of large data sets, analyzing some of the previous works published in the literature. Conceptual clustering is examined in terms of two existent applications, namely, Schema Integration and Query Monitoring. Finally, future directions on learning from databases are considered.

1.1 Paper Organization

This paper is organized as follows. Section 2 formally defines Knowledge Discovery, presenting algorithms, different approaches in the literature and possible applications of the paradigm. Section 3 covers classification methods and contains the details of conceptual clustering techniques in database systems. Section 4 summarizes the presented work and section 5 proposes some directions for future work.

2 Knowledge Discovery

Formally, we can think of knowledge discovery as the activity of finding patterns P expressed as statements S in a language L, that arise in a large Fact Base DB. Measures of certainty C may be used to encompass the reliability of the patterns discovered. A pattern that is considered interesting (defined by users) and certain enough (probabilistic criteria) may be considered **Knowledge.** The facts are stored in large Relational Databases or Object Oriented Databases.

Several approaches to find regularities within the data appear in the literature [1, 2, 9, 14]. The most popular approach is through **Classification**, where facts stored in DB are compared to produce categories. What determines category membership is some essential property which can be expressed as a small, single set of necessary and sufficient conditions. Queries to DBMS may also be monitored and classified in terms of concept hierarchies [14], providing important suggestions about possible schema transformations to database administrators.

Another approach to mining in databases is to search for **Association Rules** between sets of items. This method is also guided by user defined criteria (defining interesting patterns) and by background knowledge of the problem domain. By means of the analysis of past transaction data, relevant associations may be found, increasing the quality of the decision making task and the functionality of the DBMS.

2.1 Facets of Discovered Knowledge

Three important aspects characterize the discovered knowledge [10]:

- The **Form** of discovered knowledge can be categorized by the type of data patterns described. *Interfield Patterns* relate values of fields in the same record and *Inter-record Patterns* relate patterns aggregated over groups of records. Referring to descriptive capacity, a *Quantitative* discovery relates numeric field values of equations while a *Qualitative* discovery expresses a logical relationship among fields.

- The **Representation** of discovered knowledge may be chosen according to the intended target(s) of the discovery. For humans, the best choices are visual formalisms, natural language expressions or logic formulas. If the discoveries are fed back into the system, production rules may be an adequate formalism. In [9], the knowledge discovered can be expressed as relational tables.

- The **Uncertainty** of discovered knowledge reflects the probabilistic nature of the regularities encountered. Sampling may also be employed when large databases are accessed, decreasing even more the accuracy of the patterns found. Capturing this sort of probabilistic information may impose an extension of the logical representations with probabilistic weights.

2.2 Knowledge Discovery Algorithms

Knowledge may be extracted from data using different procedures of Knowledge Discovery. Some machine learning algorithms presented in the literature have been modified so as to be coupled to databases. Two examples are UNIMEM and COBWEB [14].

The discovery activity involves two processes: identifying interesting patterns and describing them in a concise and meaningful manner [10]. The identification process clusters conceptually coherent objects into classes. The descriptive task, in turn, summarizes relevant qualities of the identified classes.

Limitant factors of machine learning algorithms for Knowledge Discovery in Databases are: [14]:

1. The system must learn the necessary concepts through observation, by examining instances of the concepts.

2. The learning algorithm must work exclusively with positive examples as input.

3. Some algorithms must be able to work incrementally and learn concepts as examples arrive.

4. The learning algorithm must be able to form its own classifications.

5. The learning algorithm must be capable of learning multiple concepts simultaneously.

6. The learning algorithm must cope with the possibility of classifying a single instance in different concepts.

2.3 Different Approaches of Discovery in Databases

The first efforts on using machine learning algorithms in database systems dealt with issues in database design. The system learns from the encountered exceptional data objects, i.e, those that do not conform to the logical schema of the database, and suggests modifications of the schema that will accommodate them appropriately. In [16], machine learning techniques are used to address the problem of object flavor evolution in object-oriented database systems.

In [4], conceptual clustering algorithms are applied to perform Schema Integration. The integration occurs as a result of conceptual clustering the underlying data instances of different databases and guiding the process by specifying a clustering seed.

In the approach of [14], the machine learning algorithms monitor the stream of incoming queries, generating hierarchies with the most important concepts expressed in the queries. The usefulness of the hierarchies consists of showing concepts that when incorporated to the physical or external schemas of the database, may help to increase the systems performance.

The work of [2] develops an algorithm for mining association rules between items in past transactions of large databases. Support constraints are considered as to extract only relevant rules, which in turn have confidence factors associated in order to atest their reliability. Issues in buffer management and pruning techniques are also considered.

A data model for exploratory database activities was developed in [7] where description logics (DL) has been used to provide classification and propagation inference capabilities. The DL reasoner may be coupled to a DBMS providing an important interface to knowledge mining [5].

2.4 Main Areas of Research in Knowledge Mining

Knowledge mining is currently being applied to two scientific areas of research: **Chemistry** and **Biology**. In the field of Chemistry, unsupervised machine learning algorithms are being developed to extract chemical knowledge from reaction databases.

BRANGANE [20] is a system developed to extract knowledge from reaction databases in terms of graph rewriting rules. The system is based on conceptual clustering and inductive generalization methods. It works by applying a classification that performs a partitioning of the reactions in terms of three specific properties:

1. The basic structural transformation of the reaction (reaction center).

2. The requirements of the reaction.

3. What is allowed by the reaction.

In Biology, the use of machine learning algorithms in DNA sequence databases is an active area of research. In [12], regulatory features are induced by the use of Case Based Reasoning (CBR) Techniques that build classification hierarchies that reflect the evolutionary relationship between genes.

A gramatical model is used for the gene structure, where each gene is described as an instance grammar. The CBR Algorithms induce descriptive grammars of gene classes from the instance grammars.

3 Classification Methods

Classification is the endeavour of abstracting instances ocurring in the domain of discourse of an application and grouping them into classes or concepts. It is traditionally performed by database designers, which develop a conceptual model of an application, by identifying classes (object oriented databases), entity types (semantic data models) or relations (relational databases).

The classification task may be done automatically by means of machine learning techniques [9, 4, 1, 14, 13], where instances and queries to the system are compared in terms of its conceptual structures and grouped into similar classes (conceptual clustering). Other approaches to classification [5, 7] apply deductive classification to instances and descriptions using Description Logics [6] reasoners coupled to existing databases.

3.1 Conceptual Clustering

Conceptual Clustering is a machine learning technique that addresses the problem of learning by observation. In contrast to learning from examples, where the goal is to induce a description of a concept from examples thereof, in conceptual clustering, the goal is to generate classes and assign instances to them by the clustering procedure.

Important guidelines in clustering systems are:

- Clustering must be guided by a goal, purpose or context.
- Exceptions must be detected and accomodated through schema modification and schema evolution.
- Classes discovered must resemble real world concepts.

3.2 Schema Integration by Conceptual Clustering

Schema integration occurs as a result of conceptual clustering of data instances of different databases, guiding this process by specifying a clustering seed. The background knowledge required consists of:

- Set-subset rèlationship: $subset(attr_j, attr_k) \equiv dom(attr_j) \subset dom(attr_k)$
- Synonymity: $synonym(attr_j, attr_k) \equiv ObviousMeaning.$
- Logical Implication: $logical(attr_j, attr_k) \equiv attr_k \rightarrow attr_j$

The clustering seed represents the type of view desired by the user. It specifies the relevant attributes that will guide the clustering process.

SELECT $attr_j \vee attr_k$

FROM Database-1, Database-2

BY Types of Generalizations.

In the **SELECT** clause, the relevant attributes are specified. The **FROM** clause indicates the databases that are submitted to the clustering system and the **BY** declaration indicates the types of generalizations to be held on the attributes.

General steps of the clustering algorithm presented in [4] are:

1. Determine Relevant Instance Set (RIS) and Relevant Attribute Set (RAS)
2. Generate the Class Taxonomy (CT) and Derived Attribute Set (DAS)
   (a) Determine the largest common subexpressions of attributes along RIS and add attributes to DAS. A class with these attributes is created.
   (b) For each class generated C, determine the largest non-overlapping subexpressions of attributes, not members of DAS, among instances of that class. A candidate class is generated for each of the subexpressions. Instances with these subexpressions are members of the corresponding candidate classes.
   (c) Repeat step 2 until all the attributes in RAS also belong to DAS.

The first step is guided by direct matching of attributes expressed in the clustering seed and by attributes derived by applying transformations and/or inferences to the attributes. At the end of the entire process, the DAS and class taxonomy will be completely specified.

3.3 Concept Formation By Query Monitoring

In this subsection, an example of the use of a modified version of the UNIMEM algorithm [14] is presented. A stream of incoming queries to a database system is monitored, and hierarchies with the most important concepts expressed in the queries are generated by the algorithm.

Consider the straightforward schema of a Company database:

EMP(eno,ename,age,salary)

DEPT(dno,dname,floor,mgrno)

Consider also a stream of queries to the system:

Select ename
From EMP,DEPT
Where edno = dno

```
Select ename
From EMP,DEPT
Where edno = dno and age = 20 and salary = 200

Select ename
From EMP,DEPT
Where floor = edno and age = 20

Select ename
From EMP,DEPT
Where edno = dno and age = 20

Select ename
From EMP,DEPT
Where edno = dno and age = 20 and salary = 200
```

After the five queries are submited to the system, the concept hierarchy below is generated ($Concept - 4 \rightarrow Concept - 3 \rightarrow Concept - 2 \rightarrow Concept - 1$). The numbers in brackets represent the algorithm's confidence values for those feature-value combinations. The set of queries form a linear hierarchy in which the concepts become more specialized deeper in the hierarchy.

The first concept in the root of the hierarchy (concept-1) represents the third query, which could not be clustered because no matchings with other queries were found.

Concept-2 represents de notion of *employment* expressed by the join where edno = dno. Concept-3 represent the concept of *young employees* by including the feature age = 20. Finally, Concept-4 adds another feature to the above, namely salary = 200, to represent the notion of *well-paid employees.*

Concept-1
Features (F): None

Concept-2
F : edno = dno [4]

Concept-3
F : edno = dno [3] age = 20

Concept-4

F : edno = dno [2]
age = 20
salary = 200

The concept hierarchies may indicate valuable modifications to the physical and external schemas. As an example, any attributes that appear in the higher levels of the hierarchy are good candidates for indices.

The specific form in which an attribute appears in a concept indicates the appropriate type of index: if the candidate attribute appears in most concepts in equality selections and joins, hashing is to be preferred; if it appears in many concepts in nonequality selections, a B+ tree is to be preferred [14].

From a perspective of logical database design, concepts that appear in the topmost levels of the hierarchy with high confidence values may be efficiently captured by views.

4 Summary

In this paper, knowledge discovery in databases was analyzed. Initially, machine learning techniques were presented, considering different strategies, the complexity of the learning process and the representation of the discovered patterns.

Following, general aspects of knowledge discovery were presented, considering algorithms, different approaches encountered in the literature and main areas of research.

Finally, classification and conceptual clustering were analyzed. Two examples of the application of the paradigm were shown, namely, schema integration and concept formation by query monitoring. A comparative table between some of the published works is presented in the next page. The works are compared according to the learning technique(s) employed, the mining system's applicability and the mining approach adopted.

5 Research Directions

Several issues related to knowledge mining in databases have not been adequately studied and require further research. Some of the issues are:

- **Performance.** Pruning techniques and hill climbing heuristics need to be further studied and adapted, in order to cope with the complexity of time and space inherent to the database mining methods.

- **Interfaces.** Environments for interactive mining in databases are necessary to provide an adequate interface to users. Visual query languages and Browsing techniques represent the most basic features needed.

- **Architectures.** New Architectures are required, allowing supervised learning methods and parallel models of computation.

- **New Domains.** Besides Biology, Chemistry and Industrial applications, other fields like Geography, Archeology, Medicine and Agriculture may also be benefited by the use of knowledge discovery methods applied to databases.

As a future work goal, the author intends to apply conceptual clustering methods to monitor queries submited to DNA sequence databases. The idea is to obtain hierarchies from the learning system that can reflect possible modifications to the physical and external schemas to enhance the system's performance.

Reference	Learning Tech.	Applicability	Mining Approach
[9]	by Observation (Inductive Generalization)	Characteristic rules, discrimination rules, data evolution regularities	Generalization by Attribute Induction
[4]	by Observation (conceptual clustering)	Schema Integration	Classification
[14]	by Observation (conceptual clustering)	Concept Formation by Query Monitoring	Classification
[1]	from Examples	Generation of classification functions for efficient retrieval	Classification
[2]	by Observation	Examines past transaction data and generates association rules	Association
[5]	Deductive Classification	Generate Class Taxonomy for a database	Classification

References

[1] R. Agrawal, S. Ghosh, T. Imielinski, B. Iyer, A. Swami. *An Interval Classifier for Database Mining Applications.* In Proceedings of the 18th VLDB Conference, pages 560-573, Vancouver, British Columbia, Canada 1992.

[2] Rakesh Agrawal, Tomasz Imielinski, Arun Swami. *Mining Association Rules Between Sets of Items in Large Databases.* In Proceedings of ACM SIGMOD International Conference on Management of Data, pages 207-216, Washington, DC, May 1993.

[3] D. E. Altenkrueger. *KBMS: Aspects, Theory and Implementation.* Information Systems, Vol 15, 1, 1990.

[4] H. W. Beck, T. M. Anwar, S. B. Navathe. *Classification Through Conceptual Clustering in Database Systems.* Proceedings of 1st International Conference on Information and Knowledge Management, pages 465-472, Baltimore, MD, November, 1992.

[5] Alex Borgida, Ronald J. Brachman. *Loading Data into Description Reasoners.* In Proceedings of ACM SIGMOD International Conference on Management of Data, pages 217-226, Washington, DC, May 1993.

[6] Alex Borgida. *Descrption Logics are not just for the Flightless-Birds: A New Look at the Utility and Foundations of Description Logics.* Computer Science Technical Report, Rutgers University, June 1992.

[7] Alex Borgida, Ronald Brachman, Deborah McGuinness, Lori Resnick. *CLASSIC: A Structural Data Model For Objects.* In Proceedings of ACM SIGMOD International Conference on Management of Data, pages 58-67, May 1989.

[8] J. G. Carbonell, R. S. Michalski, T. M. Mitchell. *An Overview of Machine Learning* in [18].

[9] Jiawei Han, Yandong Cai, Nick Cercone. *Knowledge Discovery in Databases: An Attribute-Oriented Approach.* In Proceedings of the 18th VLDB Conference, pages 547-559, Vancouver, British Columbia, Canada 1992.

[10] W. J. Frawley, G. Piatetsky-Shapiro, C. J. Matheus. *Knowledge Discovery in Databases: An Overview.* In [15].

[11] J. H. Gennari, P. Langley, D. Fisher. *Models of Incremental Concept Formation.* Artificial Intelligence, 40, 1989.

[12] J. Haas, J. S. Aaronson and C. Overton. *Analogical Reasoning for Knowledge Discovery in a Molecular Biology Database.* In Proceedings of the 2nd International Conference in Knowledge Management, pages 554-564, Washington, DC, November 1993.

[13] Jiawei Han, Yandong Cai, Nick Cercone. *DBLEARN: A Knowledge Discovery System for Large Databases.* Proceedings of 1st International Conference on Information and Knowledge Management, pages 473-481, Baltimore, MD, November, 1992.

[14] Yannis E. Ioannidis, Tomas Saulys, Andrew J. Whitsitt. *Conceptual Learning in Database Systems.* ACM Transactions on Information Systems, Vol. 10, No. 3, July 1992.

[15] Gregory Piatetsky-Shapiro, William J. Frawley. *Knowledge Discovery in Databases.* AIII / MIT Press, 1991.

[16] Q. Li, D. McLeod. *Object Flavor Evolution through Learning in an Object Oriented Database System.* In Expert Database Systems, Proceedings from the Second International Conference, pages 469-495, Benjaming / Cummings, Menlo Park, Calif. 1989.

[17] Ryszard S. Michalski. *Learning Strategies and Automated Knowledge Acquisition. An Overview.* In Computational Models of Learning, edited by Leonard Bolc, Springer-Verlag, 1987.

[18] R. S. Michalski, J. G. Carbonell, T. M. Mitchell. *Machine Learning: An Artificial Intelligence Approach.* Tioga Publishing Co., Palo Alto, CA, 1983.

[19] Jeniffer Widom, Eric Hansom. *An Overview of Production Rules in Database Systems.* RJ 9023 (80483), IBM Research Division, Almaden, October 12, 1992.

[20] J. Royce and Herbert Gelernter. *Knowledge Discovery in Reaction Databases.* Proceedings of the 2nd International Conference on Knowledge Management, pages 714-716, Washington, DC, November 1993.

[21] A. Silberschatz, M. Stonebraker, J. Ullman. *Database Systems: Achievements and Opportunities.* CACM, October 1991.

[22] J.D. Ullman. *Principles of Database and Knowledge-Base Systems, Vol I.* Computer Science Press, Rockville, Maryland, 1989.

[23] L.G. Valiant. *A theory of the Learnable.* CACM, November 1984.