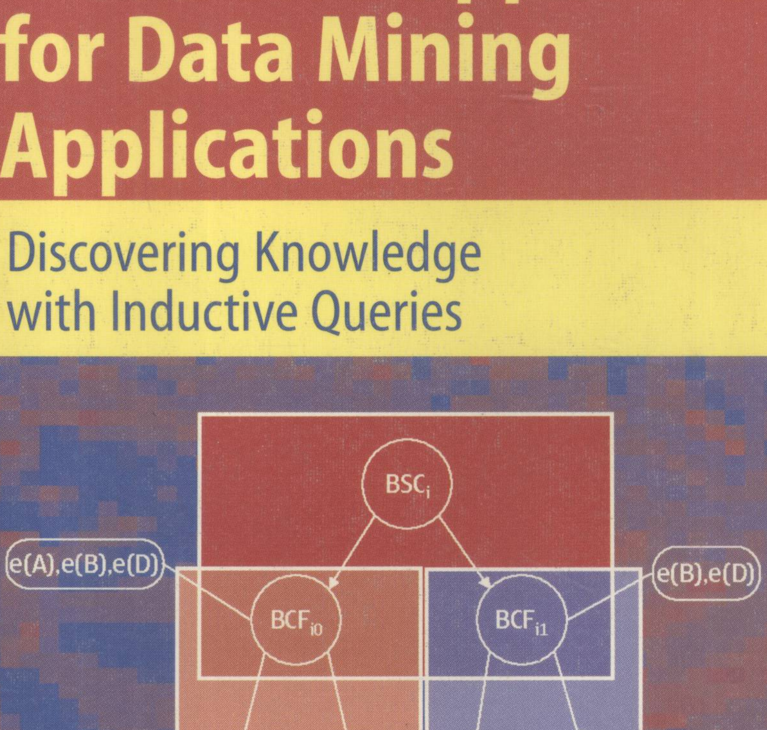
State-of-the-Art Survey

LNAI 2682

Rosa Meo
Pier Luca Lanzi
Mika Klemettinen (Eds.)

Database Support for Data Mining Applications

Discovering Knowledge with Inductive Queries



Springer

Rosa Meo   Pier Luca Lanzi
Mika Klemettinen (Eds.)

Database Support for Data Mining Applications

Discovering Knowledge with Inductive Queries

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Rosa Meo
Università degli Studi di Torino
Dipartimento di Informatica
Corso Svizzera, 185, 10149 Torino, Italy
E-mail: meo@di.unito.it

Pier Luca Lanzi
Politecnico di Milano
Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci 32, 20133 Milano, Italy
E-mail: lanzi@elet.polimi.it

Mika Klemettinen
Nokia Research Center
P.O.Box 407, Itämerenkatu 11-13, 00045 Nokia Group, Finland
E-mail: mika.klemettinen@nokia.com

Library of Congress Control Number: 2004095564

CR Subject Classification (1998): I.2, H.2, H.3

ISSN 0302-9743
ISBN 3-540-22479-3 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag is a part of Springer Science+Business Media

springeronline.com

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11301172 06/3142 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 2682

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Preface

Data mining from traditional relational databases as well as from non-traditional ones such as semi-structured data, Web data and scientific databases such as biological, linguistic and sensor data has recently become a popular way of discovering hidden knowledge. In the context of relational and traditional data, methods such as association rules, chi square rules, ratio rules, implication rules, etc. have been proposed in multiple, varied contexts. In the context of non-traditional data, newer, more experimental yet novel techniques are being proposed. There is an agreement among the researchers across communities that data mining is a key ingredient for success in their respective areas of research and development. Consequently, interest in developing new techniques for data mining has peaked and a tremendous stride is being made to answer interesting and fundamental questions in various disciplines using data mining.

In the past, researchers mainly focused on algorithmic issues in data mining and placed much emphasis on scalability. Recently, the focus has shifted towards a more declarative way of answering questions using data mining that has given rise to the concept of mining queries.

Data mining has recently been applied with success to discovering hidden knowledge from relational databases. Methods such as association rules, chi square rules, ratio rules, implication rules, etc. have been proposed in several and very different contexts. To cite just the most frequent and famous ones: the market basket analysis, failures in telecommunication networks, text analysis for information retrieval, Web content mining, Web usage, log analysis, graph mining, information security and privacy, and finally analysis of objects traversal by queries in distributed information systems.

From these widespread and various application domains it results that data mining rules constitute a successful and intuitive descriptive paradigm able to offer complementary choices in rule induction. Other than inductive and abductive logic programming, research into data mining from knowledge bases has been almost non-existent, because contemporary methods place the emphasis on the scalability and efficiency of algorithmic solutions, whose inherent procedurality is difficult to cast into the declarativity of knowledge base systems.

In particular, researchers convincingly argue that the ability to declaratively mine and analyze relational databases for decision support is a critical requirement for the success of the acclaimed data mining technology. Indeed, DBMSs constitute today one of the most advanced and sophisticated achievements that applied computer science has made in the past years. Unfortunately, almost all the most powerful DBMSs we have today have been developed with a focus on On-Line Transaction-Processing tasks. Instead, database technology for On-Line Analytical-Processing tasks, such as data mining, is more recent and in need of further research.

Although there have been several encouraging attempts at developing methods for data mining using SQL, simplicity and efficiency still remain significant prerequisites for further development. It is well known that today database technology is mature enough: popular DBMSs, such as Oracle, DB2 and SQL-Server, provide interfaces, services, packages and APIs that embed data mining algorithms for classification, clustering, association rules extraction and temporal sequences, such that they are directly available to programmers and ready to be called by applications.

Therefore, it is envisioned that we should be able now to mine relational databases for interesting rules directly from database query languages, without any data restructuring or preprocessing steps. Hence no additional machineries with respect to database languages would be necessary. This vision entails that the optimization issues should be addressed at the system level for which we have now a significant body of research, while the analyst could concentrate better on the declarative and conceptual level, in which the difficult task of interpretation of the extracted knowledge occurs. Therefore, it is now time to develop declarative paradigms for data mining so that these developments can be exploited at the lower and system level, for query optimization.

With this aim we planned this book on "Data Mining" with an emphasis on approaches that exploit the available database technology, declarative data mining, intelligent querying and associated issues such as optimization, indexing, query processing, languages and constraints. Attention is also paid to solution of data preprocessing problems, such as data cleaning, discretization and sampling, developed using database tools and declarative approaches, etc.

Most of this book resulted also as a consequence of the work we conducted during the development of the *cInQ* project (consortium on discovering knowledge with **In**ductive **Q**ueries) an EU funded project (IST 2000-26469) aiming at developing database technology for leveraging decision support systems by means of query languages and inductive approaches to knowledge extraction from databases. It presents new and invited contributions, plus the best papers, extensively revised and enlarged, presented during workshops on the topics of database technology, data mining and inductive databases at international conferences such as EDBT and PKDD/ECML, in 2002.

May 2004

Rosa Meo
Pier Luca Lanzi
Mika Klemettinen

Volume Organization

This volume is organized in two main sections. The former focuses on *Database Languages and Query Execution*, while the latter focuses on methodologies, techniques and new approaches that provide *Support for Knowledge Discovery Process*. Here, we briefly overview each contribution.

Database Languages and Query Execution

The first contribution is *Inductive Databases and Multiple Uses of Frequent Itemsets: The cInQ Approach* which presents the main contributions of theoretical and applied nature, in the field of inductive databases obtained in the `cInQ` project.

In *Query Languages Supporting Descriptive Rule Mining: A Comparative Study* we provide a comparison of features of available relational query languages for data mining, such as `DMQL`, `MSQL`, `MINE RULE`, and standardization efforts for coupling database technology and data mining systems, such as `OLEDB-DM` and `PMML`.

Declarative Data Mining Using SQL-3 shows a new approach, compared to existing SQL approaches, to mine association rules from an object-relational database: it uses a recursive join in SQL-3 that allows no restructuring or preprocessing of the data. It proposes a new `mine by` SQL-3 operator for capturing the functionality of the proposed approach.

Towards a Logic Query Language for Data Mining presents a logic database language with elementary data mining mechanisms, such as user-defined aggregates that provide a model, powerful and general as well, of the relevant aspects and tasks of knowledge discovery.

Data Mining Query Language for Knowledge Discovery in a Geographical Information System presents *SDMOQL* a spatial data mining query language for knowledge extraction from GIS. The language supports the extraction of classification rules and association rules, the use of background models, various interestingness measures and the visualization.

Towards Query Evaluation in Inductive Databases Using Version Spaces studies inductive queries. These ones specify constraints that should be satisfied by the data mining patterns in which the user is interested. This work investigates the properties of solution spaces of queries with monotonic and anti-monotonic constraints and their boolean combinations.

The GUHA Method, Data Preprocessing and Mining surveys the basic principles and foundations of the *GUHA* method, the available systems and related works. This method originated in the Czechoslovak Academy of Sciences of Prague in the mid 1960s with strong logical and statistical foundations. Its main principle is to let the computer generate and evaluate all the hypotheses that may be interesting given the available data and the domain problem. This work discusses also the relationships between the GUHA method and relational data mining and discovery science.

Constraint Based Mining of First Order Sequences in SeqLog presents a logical language, *SeqLog*, for mining and querying sequential data and databases. This language is used as a representation language for an inductive database system. In this system, variants of level-wise algorithms for computing the version space of the solutions are proposed and experimented in the user-modeling domain.

Support for Knowledge Discovery Process

Interactivity, Scalability and Resource Control for Efficient KDD Support in DBMS proposes a new approach for combining preprocessing and data mining operators in a KDD-aware implementation algebra. In this way data mining operators can be integrated smoothly into a database system, thus allowing interactivity, scalability and resource control. This framework is based on the extensive use of pipelining and is built upon an extended version of a specialized database index.

Frequent Itemset Discovery with SQL Using Universal Quantification investigates the integration of data analysis functionalities into two basic components of a database management system: query execution and optimization. It employs universal and existential quantifications in queries and a vertical layout to ease the set containment operations needed for frequent itemsets discovery.

Deducing Bounds on the Support of Itemsets provides a complete set of rules for deducing tight bounds on the support of an itemset if the support of all its subsets are known. These bounds can be used by the data mining system to choose the best access path to data and provide a better representation of the collection of frequent itemsets.

Model-Independent Bounding of the Supports of Boolean Formulae in Binary Data considers frequencies of arbitrary boolean formulas, a new class of aggregates: the summaries. These ones are computed for descriptive purposes on a sparse binary data set. This work considers the problem of finding tight upper bounds on these frequencies and gives a general formulation of the problem with a linear programming solution.

Condensed Representations for Sets of Mining Queries proposes a general framework for condensed representations of sets of mining queries, defined by monotonic and anti-monotonic selection predicates. This work proves important for inductive and database systems for data mining since it deals with *sets* of queries, whereas previous work in maximal, closed and condensed representations treated so far the representation of a *single* query only.

Acknowledgments

For reviewing the contributions included in this volume we relied on a group of internationally well-known experts who we wish to thank here.

Roberto Bayardo	IBM Almaden Research Center, USA
Elena Baralis	Politecnico di Torino, Italy
Christian Böhm	University for Health Informatics and Technology, Austria
Saso Dzeroski	Jozef Stefan Institute, Slovenia
Kris Koperski	Insightful Corporation, USA
Stefan Kramer	Technische Universität München, Germany
Dominique Laurent	Université F. Rabelais, Tours, France
Giuseppe Manco	ICAR-CNR, Italy
Stefano Paraboschi	Università di Bergamo, Italy
Jian Pei	The State University of New York, USA
Giuseppe Psaila	Università di Bergamo, Italy
Lorenza Saitta	Università del Piemonte Orientale, Italy
Maria Luisa Sapino	Università di Torino, Italy
Hannu Toivonen	University of Helsinki, Finland
Jiong Yang	University of Illinois at Urbana Champaign, USA
Mohammed Zaki	Rensselaer Polytechnic Institute, USA

One-Sided Instance-Based Boundary Sets introduces a family of version-space representations that are important for their applicability to inductive databases. They correspond to the task of concept learning from a database of examples when this database is updated. One-sided instance-based boundary sets are shown to be correctly and efficiently computable.

Domain Structures in Filtering Irrelevant Frequent Patterns introduces a notion of domain constraints, based on distance measures and in terms of domain structure and concept taxonomies. Domain structures are useful in the analysis of communications networks and complex systems. Indeed they allow irrelevant combinations of events that reflect the simultaneous information of independent processes in the same database to be pruned.

Integrity Constraints over Association Rules investigates the notion of integrity constraints in inductive databases. This concept is useful in detecting inconsistencies in the results of common data mining tasks. This work proposes a form of integrity constraints called *association map constraints* that specifies the allowed variations in confidence and support of association rules.

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 3139: F. Iida, R. Pfeifer, L. Steels, Y. Kuniyoshi (Eds.), Embodied Artificial Intelligence. IX, 331 pages. 2004.

Vol. 3131: V. Torra, Y. Narukawa (Eds.), Modeling Decisions for Artificial Intelligence. XI, 327 pages. 2004.

Vol. 3127: K.E. Wolff, H.D. Pfeiffer, H.S. Delugach (Eds.), Conceptual Structures at Work. XI, 403 pages. 2004.

Vol. 3123: A. Belz, R. Evans, P. Piwek (Eds.), Natural Language Generation. X, 219 pages. 2004.

Vol. 3120: J. Shawe-Taylor, Y. Singer (Eds.), Learning Theory. X, 648 pages. 2004.

Vol. 3097: D. Basin, M. Rusinowitch (Eds.), Automated Reasoning. XII, 493 pages. 2004.

Vol. 3071: A. Omicini, P. Petta, J. Pitt (Eds.), Engineering Societies in the Agents World. XIII, 409 pages. 2004.

Vol. 3070: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing - ICAISC 2004. XXV, 1208 pages. 2004.

Vol. 3068: E. André, L. Dybkjær, W. Minker, P. Heisterkamp (Eds.), Affective Dialogue Systems. XII, 324 pages. 2004.

Vol. 3067: M. Dastani, J. Dix, A. El Fallah-Seghrouchni (Eds.), Programming Multi-Agent Systems. X, 221 pages. 2004.

Vol. 3066: S. Tsumoto, R. Słowiński, J. Komorowski, J.W. Grzymała-Busse (Eds.), Rough Sets and Current Trends in Computing. XX, 853 pages. 2004.

Vol. 3065: A. Lomuscio, D. Nute (Eds.), Deontic Logic in Computer Science. X, 275 pages. 2004.

Vol. 3060: A.Y. Tawfik, S.D. Goodwin (Eds.), Advances in Artificial Intelligence. XIII, 582 pages. 2004.

Vol. 3056: H. Dai, R. Srikant, C. Zhang (Eds.), Advances in Knowledge Discovery and Data Mining. XIX, 713 pages. 2004.

Vol. 3055: H. Christiansen, M.-S. Hacid, T. Andreasen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 500 pages. 2004.

Vol. 3040: R. Conejo, M. Urretavizcaya, J.-L. Pérez-de-la-Cruz (Eds.), Current Topics in Artificial Intelligence. XIV, 689 pages. 2004.

Vol. 3035: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XII, 326 pages. 2004.

Vol. 3034: J. Favela, E. Menasalvas, E. Chávez (Eds.), Advances in Web Intelligence. XIII, 227 pages. 2004.

Vol. 3030: P. Giorgini, B. Henderson-Sellers, M. Winikoff (Eds.), Agent-Oriented Information Systems. XIV, 207 pages. 2004.

Vol. 3029: B. Orchard, C. Yang, M. Ali (Eds.), Innovations in Applied Artificial Intelligence. XXI, 1272 pages. 2004.

Vol. 3025: G.A. Vouros, T. Panayiotopoulos (Eds.), Methods and Applications of Artificial Intelligence. XV, 546 pages. 2004.

Vol. 3020: D. Polani, B. Browning, A. Bonarini, K. Yoshida (Eds.), RoboCup 2003: Robot Soccer World Cup VII. XVI, 767 pages. 2004.

Vol. 3012: K. Kurumatani, S.-H. Chen, A. Ohuchi (Eds.), Multi-Agnets for Mass User Support. X, 217 pages. 2004.

Vol. 3010: K.R. Apt, F. Fages, F. Rossi, P. Szeredi, J. Váncza (Eds.), Recent Advances in Constraints. VIII, 285 pages. 2004.

Vol. 2990: J. Leite, A. Omicini, L. Sterling, P. Torroni (Eds.), Declarative Agent Languages and Technologies. XII, 281 pages. 2004.

Vol. 2980: A. Blackwell, K. Marriott, A. Shimojima (Eds.), Diagrammatic Representation and Inference. XV, 448 pages. 2004.

Vol. 2977: G. Di Marzo Serugendo, A. Karageorgos, O.F. Rana, F. Zambonelli (Eds.), Engineering Self-Organising Systems. X, 299 pages. 2004.

Vol. 2972: R. Monroy, G. Arroyo-Figueroa, L.E. Sucar, H. Sossa (Eds.), MICAI 2004: Advances in Artificial Intelligence. XVII, 923 pages. 2004.

Vol. 2961: P. Eklund (Ed.), Concept Lattices. IX, 411 pages. 2004.

Vol. 2953: K. Konrad, Model Generation for Natural Language Interpretation and Analysis. XIII, 166 pages. 2004.

Vol. 2934: G. Lindemann, D. Moldt, M. Paolucci (Eds.), Regulated Agent-Based Social Systems. X, 301 pages. 2004.

Vol. 2930: F. Winkler (Ed.), Automated Deduction in Geometry. VII, 231 pages. 2004.

Vol. 2926: L. van Elst, V. Dignum, A. Abecker (Eds.), Agent-Mediated Knowledge Management. XI, 428 pages. 2004.

Vol. 2923: V. Lifschitz, I. Niemelä (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 365 pages. 2004.

Vol. 2915: A. Camurri, G. Volpe (Eds.), Gesture-Based Communication in Human-Computer Interaction. XIII, 558 pages. 2004.

Vol. 2913: T.M. Pinkston, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2003. XX, 512 pages. 2003.

Vol. 2903: T.D. Gedeon, L.C.C. Fung (Eds.), AI 2003: Advances in Artificial Intelligence. XVI, 1075 pages. 2003.

Vol. 2902: F.M. Pires, S.P. Abreu (Eds.), Progress in Artificial Intelligence. XV, 504 pages. 2003.

Vol. 2892: F. Dau, The Logic System of Concept Graphs with Negation. XI, 213 pages. 2003.

Vol. 2891: J. Lee, M. Barley (Eds.), Intelligent Agents and Multi-Agent Systems. X, 215 pages. 2003.

Vol. 2882: D. Veit, Matchmaking in Electronic Markets. XV, 180 pages. 2003.

Vol. 2871: N. Zhong, Z.W. Raś, S. Tsumoto, E. Suzuki (Eds.), Foundations of Intelligent Systems. XV, 697 pages. 2003.

Vol. 2854: J. Hoffmann, Utilizing Problem Structure in Planing. XIII, 251 pages. 2003.

Vol. 2843: G. Grieser, Y. Tanaka, A. Yamamoto (Eds.), Discovery Science. XII, 504 pages. 2003.

Vol. 2842: R. Gavaldá, K.P. Jantke, E. Takimoto (Eds.), Algorithmic Learning Theory. XI, 313 pages. 2003.

Vol. 2838: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Knowledge Discovery in Databases: PKDD 2003. XVI, 508 pages. 2003.

Vol. 2837: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Machine Learning: ECML 2003. XVI, 504 pages. 2003.

Vol. 2835: T. Horváth, A. Yamamoto (Eds.), Inductive Logic Programming. X, 401 pages. 2003.

Vol. 2821: A. Günter, R. Kruse, B. Neumann (Eds.), KI 2003: Advances in Artificial Intelligence. XII, 662 pages. 2003.

Vol. 2807: V. Matoušek, P. Mautner (Eds.), Text, Speech and Dialogue. XIII, 426 pages. 2003.

Vol. 2801: W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, J.T. Kim (Eds.), Advances in Artificial Life. XVI, 905 pages. 2003.

Vol. 2797: O.R. Zaïane, S.J. Simoff, C. Djeraba (Eds.), Mining Multimedia and Complex Data. XII, 281 pages. 2003.

Vol. 2792: T. Rist, R.S. Aylett, D. Ballin, J. Rickel (Eds.), Intelligent Virtual Agents. XV, 364 pages. 2003.

Vol. 2782: M. Klusch, A. Omicini, S. Ossowski, H. Laamanen (Eds.), Cooperative Information Agents VII. XI, 345 pages. 2003.

Vol. 2780: M. Dojat, E. Keravnou, P. Barahona (Eds.), Artificial Intelligence in Medicine. XIII, 388 pages. 2003.

Vol. 2777: B. Schölkopf, M.K. Warmuth (Eds.), Learning Theory and Kernel Machines. XIV, 746 pages. 2003.

Vol. 2752: G.A. Kaminka, P.U. Lima, R. Rojas (Eds.), RoboCup 2002: Robot Soccer World Cup VI. XVI, 498 pages. 2003.

Vol. 2741: F. Baader (Ed.), Automated Deduction – CADE-19. XII, 503 pages. 2003.

Vol. 2705: S. Renals, G. Grefenstette (Eds.), Text- and Speech-Triggered Information Access. VII, 197 pages. 2003.

Vol. 2703: O.R. Zaïane, J. Srivastava, M. Spiliopoulou, B. Masand (Eds.), WEBKDD 2002 - MiningWeb Data for Discovering Usage Patterns and Profiles. IX, 181 pages. 2003.

Vol. 2700: M.T. Pazienza (Ed.), Extraction in the Web Era. XIII, 163 pages. 2003.

Vol. 2699: M.G. Hinchey, J.L. Rash, W.F. Truszkowski, C.A. Rouff, D.F. Gordon-Spears (Eds.), Formal Approaches to Agent-Based Systems. IX, 297 pages. 2002.

Vol. 2691: V. Mařík, J.P. Müller, M. Pechoucek (Eds.), Multi-Agent Systems and Applications III. XIV, 660 pages. 2003.

Vol. 2684: M.V. Butz, O. Sigaud, P. Gérard (Eds.), Anticipatory Behavior in Adaptive Learning Systems. X, 303 pages. 2003.

Vol. 2682: R. Meo, P.L. Lanzi, M. Klemettinen (Eds.), Database Support for Data Mining Applications. XII, 325 pages. 2004.

Vol. 2671: Y. Xiang, B. Chaib-draa (Eds.), Advances in Artificial Intelligence. XIV, 642 pages. 2003.

Vol. 2663: E. Menasalvas, J. Segovia, P.S. Szczepaniak (Eds.), Advances in Web Intelligence. XII, 350 pages. 2003.

Vol. 2661: P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. VII, 231 pages. 2003.

Vol. 2654: U. Schmid, Inductive Synthesis of Functional Programs. XXII, 398 pages. 2003.

Vol. 2650: M.-P. Huget (Ed.), Communications in Multiagent Systems. VIII, 323 pages. 2003.

Vol. 2645: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XI, 320 pages. 2003.

Vol. 2639: G. Wang, Q. Liu, Y. Yao, A. Skowron (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. XVII, 741 pages. 2003.

Vol. 2637: K.-Y. Whang, J. Jeon, K. Shim, J. Srivastava, Advances in Knowledge Discovery and Data Mining. XVIII, 610 pages. 2003.

Vol. 2636: E. Alonso, D. Kudenko, D. Kazakov (Eds.), Adaptive Agents and Multi-Agent Systems. XIV, 323 pages. 2003.

Vol. 2627: B. O'Sullivan (Ed.), Recent Advances in Constraints. X, 201 pages. 2003.

Vol. 2600: S. Mendelson, A.J. Smola (Eds.), Advanced Lectures on Machine Learning. IX, 259 pages. 2003.

Vol. 2592: R. Kowalczyk, J.P. Müller, H. Tianfield, R. Unland (Eds.), Agent Technologies, Infrastructures, Tools, and Applications for E-Services. XVII, 371 pages. 2003.

Vol. 2586: M. Klusch, S. Bergamaschi, P. Edwards, P. Petta (Eds.), Intelligent Information Agents. VI, 275 pages. 2003.

Vol. 2583: S. Matwin, C. Sammut (Eds.), Inductive Logic Programming. X, 351 pages. 2003.

Vol. 2581: J.S. Sichman, F. Bousquet, P. Davidsson (Eds.), Multi-Agent-Based Simulation. X, 195 pages. 2003.

Vol. 2577: P. Petta, R. Tolksdorf, F. Zambonelli (Eds.), Engineering Societies in the Agents World III. X, 285 pages. 2003.

Vol. 2569: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. XIII, 648 pages. 2002.

Vol. 2560: S. Goronzy, Robust Adaptation to Non-Native Accents in Automatic Speech Recognition. XI, 144 pages. 2002.

Vol. 2557: B. McKay, J. Slaney (Eds.), AI 2002: Advances in Artificial Intelligence. XV, 730 pages. 2002.

Vol. 2554: M. Beetz, Plan-Based Control of Robotic Agents. XI, 191 pages. 2002.

Table of Contents

I Database Languages and Query Execution

II Support for KDD-Process

Inductive Databases and Multiple Uses of Frequent Itemsets: The cINQ Approach

Jean-François Boulicaut

Institut National des Sciences Appliquées de Lyon,
LIRIS CNRS FRE 2672, Bâtiment Blaise Pascal
F-69621 Villeurbanne cedex, France

Abstract. Inductive databases (IDBs) have been proposed to afford the problem of knowledge discovery from huge databases. With an IDB the user/analyst performs a set of very different operations on data using a query language, powerful enough to perform all the required elaborations, such as data preprocessing, pattern discovery and pattern post-processing. We present a synthetic view on important concepts that have been studied within the cINQ European project when considering the pattern domain of itemsets. Mining itemsets has been proved useful not only for association rule mining but also feature construction, classification, clustering, etc. We introduce the concepts of pattern domain, evaluation functions, primitive constraints, inductive queries and solvers for itemsets. We focus on simple high-level definitions that enable to forget about technical details that the interested reader will find, among others, in cINQ publications.

1 Introduction

Knowledge Discovery in Databases (KDD) is a complex interactive process which involves many steps that must be done sequentially. In the cINQ project[1], we want to develop a new generation of databases, called "*inductive databases*" (IDBs), suggested by Imielinski and Mannila in [42] and for which a simple formalization has been proposed in [20]. This kind of databases integrate *raw data* with *knowledge* extracted from *raw data*, materialized under the form of patterns into a common framework that supports the knowledge discovery process within a database framework. In this way, the process of KDD consists essentially in a querying process, enabled by a query language that can deal either with raw data or patterns and that can be used throughout the whole KDD process across many different applications. A few query languages can be considered as candidates for inductive databases. For instance, considering the prototypical case of assoc-

[1] This research is partially funded by the Future and Emerging Technologies arm of the IST Programme FET-Open scheme (cINQ project IST-2000-26469). The author warmly acknowledges all the contributors to the cINQ project and more particularly Luc De Raedt, Baptiste Jeudy, Mika Klemettinen, and Rosa Meo.

R. Meo et al. (Eds.): Database Support for Data Mining Applications, LNAI 2682, pp. 1–23, 2004.

iation rule mining, [10] is a comparative evaluation of three proposals (MSQL [43], DMQL [38], and MINE RULE [59]) in the light of the IDBs' requirements.

In this paper, we focus on mining queries, the so-called *inductive queries*, i.e., queries that return patterns from a given database. More precisely, we consider the pattern domain of itemsets and databases that are transactional databases. Doing so, we can provide examples of concepts that have emerged as important within the cINQ project after 18 months of work.

It is useful to abstract the meaning of mining queries. A simple model has been introduced in [55] that considers a data mining process as a sequence of queries over the data but also the so-called *theory* of the data. Given a language $\mathcal{L}$ of patterns (e.g., itemsets, sequences, association rules), the theory of a database r with respect to $\mathcal{L}$ and a selection predicate q is the set $Th(r, \mathcal{L}, q) = \{\phi \in \mathcal{L} \mid q(r, \phi) \text{ is true}\}$. The predicate q indicates whether a pattern ϕ is considered interesting (e.g., ϕ denotes a property that is "frequent" in r). The selection predicate can be defined as a combination (boolean expression) of primitive constraints that have to be satisfied by the patterns. Some of them refer to the "behavior" of a pattern in the data, e.g., its "frequency" in a given data set is above or below a user-given threshold, some others define syntactical restrictions on desired patterns, e.g., its "length" is below a user-given threshold. Preprocessing concerns the definition of the database r, the mining phase is often the computation of the specified theory while post-processing can be considered as a querying activity on a materialized theory or the computation of a new theory.

This formalization however does not reflect the context of many classical data mining processes. Quite often, the user is interested not only in a collection of patterns that satisfy some constraints (e.g., frequent patterns, strong rules, approximate inclusion or functional dependencies) but also to some properties of these patterns in the selected database (e.g., their frequencies, the error for approximate dependencies). In that case, we will consider the so-called *extended theories*. For instance, when mining frequent itemsets or frequent and valid association rules [2], the user needs for the frequency of the specified patterns or rules. Indeed, during the needed post-processing phase, the user/analyst often uses various objective interestingness measures like the confidence [2], the conviction [23] or the J-mesure [72] that are computed efficiently provided that the frequency of each frequent itemset is available. Otherwise, it might be extremely expensive to look at the data again.

Designing *solvers* for more or less primitive constraints concerns the core of data mining algorithmic research. We must have solvers that can compute the (extended) theories and that have good properties in practice (e.g., scalability w.r.t. the size of the database or the size of the search space). A "generate and test" approach that would enumerate the sentences of $\mathcal{L}$ and then test the selection predicate q is generally impossible. A huge effort has concerned a clever use of the constraints occurring in q to have a tractable evaluation of useful inductive queries. This is the research area of *constraint-based data mining*. Most of the algorithmic research in pattern discovery tackles the design of complete algorithms for computing (extended) theories given more or less specific conjunctions of primitive constraints. Typically, many researchers have considered the computation of frequent patterns, i.e., patterns that satisfy a minimal frequency constraint. An important paper on a generic algorithm for such a typical mining task is [55]. However, if the active use of the so-called *anti-monotonic constraints* (e.g., the minimal frequency) is now well-understood, the situation is far less clear for non anti-monotonic constraints [64, 51, 18].

A second major issue is the possibility to approximate the results of (extended) inductive queries. This approximation can concern a collection of patterns that is a superset or a subset of the desired collection. This is the typical case when the theories are computed from a sample of the data (see, e.g., [74]) or when a relaxed constraint is used. Another important case of approximation for extended theories is the exact computation of the underlying theory while the evaluation functions are only approximated. This has lead to an important research area, the computation of the so-called *condensed representations* [54], a domain in which we have been playing a major role since the study of frequent closed itemsets as an ϵ-adequate representation for frequency queries [12].

This paper is organized as follows. Section 2 introduces notations and definitions that are needed for discussing inductive queries that return itemsets. It contains an instance of the definition of a pattern domain. Section 3 identifies several important open problems. Section 4 provides elements of solution that are currently studied within the cINQ project. Section 5 is a short conclusion.

2 A Pattern Domain for Itemsets

The definition of a *pattern domain* is made of the definition of a language of patterns $\mathcal{L}$, evaluation functions that assign a semantics to each pattern in a given database $\mathbf{r}$, languages for primitive constraints that specify the desired patterns, and inductive query languages that provide a language for combining the primitive constraints.

We do not claim that this paper is an exhaustive description of the itemset pattern domain. Even though we selected representative examples of evaluation functions and primitive constraints, many others have been or might be defined and used.

2.1 Language of Patterns and Terminology

We introduce some notations that are used for defining the pattern domain of itemsets. In that context, we consider that:

- A so-called *transactional database* contains the data,
- Patterns are the so-called *itemsets* and one kind of descriptive rule that can be derived from them, i.e., the *association rules.*

Definition 1 (Transactional Databases). *Assume that* Items *is a finite set of symbols denoted by capital letters, e.g.,* Items= $\{\texttt{A},\texttt{B},\texttt{C},\ldots\}$. *A* transaction