Studies in Computational Intelligence 471

Fabrice Guillet
Bruno Pinaud
Gilles Venturini
Djamel Abdelkader Zighed (Eds.)

Advances in Knowledge Discovery and Management

Volume 3

Springer

Studies in Computational Intelligence 471

For further volumes:
http://www.springer.com/series/7092

Fabrice Guillet, Bruno Pinaud, Gilles Venturini,
and Djamel Abdelkader Zighed (Eds.)

Advances in Knowledge Discovery and Management

Volume 3

Springer

Editors

Fabrice Guillet
LINA (CNRS UMR 6241)
Polytechnic School of Nantes University
Nantes Cedex 3
France

Bruno Pinaud
Univ. Bordeaux 1, LaBRI
Talence Cedex
France

Gilles Venturini
Université François-Rabelais de Tours
Polytech'Tours, Dpt Informatique
Tours
France

Djamel Abdelkader Zighed
Laboratoire ERIC
Université Lumiére Lyon 2
Bron
France

ISSN 1860-949X
e-ISSN 1860-9503
ISBN 978-3-642-35854-8
e-ISBN 978-3-642-35855-5
DOI 10.1007/978-3-642-35855-5
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2012955287

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The recent and novel research contributions collected in this book are extended and reworked versions of a selection of the best papers that were originally presented in French at the EGC'2011 Conference held in Brest, France, on January 2011. These 10 best papers have been selected from the 34 papers accepted in long format at the conference. These 34 long papers were themselves the result of a peer and blind review process among the 131 papers initially submitted to the conference in 2011 (acceptance rate of 26% for long papers). This conference was the 11^{th} edition of this event, which takes place each year and which is now successful and well-known in the French-speaking community. This community was structured in 2003 by the foundation of the International French-speaking EGC society (EGC in French stands for "Extraction et Gestion des Connaissances" and means "Knowledge Discovery and Management", or KDM). This society organizes every year its main conference (about 200 attendees) but also workshops and other events with the aim of promoting exchanges between researchers and companies concerned with KDM and its applications in business, administration, industry or public organizations. For more details about the EGC society, please consult http://www.egc.asso.fr.

Structure of the Book

This book is a collection of representative and novel works done in Data Mining, Knowledge Discovery, Business Intelligence, Knowledge Engineering and Semantic Web. It is intended to be read by all researchers interested in these fields, including PhD or MSc students, and researchers from public or private laboratories. It concerns both theoretical and practical aspects of KDM.

This book has been structured in two parts. The first part, entitled "Data Mining, classification and queries", deals with rule and pattern mining, with topological approaches and with OLAP. Three chapters study rule and pattern mining and concern binary data sets, sequences, and association rules. Chapters related to topological approaches study different distance measures and a new method that learns a hierarchical topological map. Finally, one chapter deals with OLAP and studies the mining of queries logs.

The second part of the book, entitled “Ontology and Semantic”, is more related to knowledge-based and user-centered approaches in KDM. One chapter deals with the enrichment of folksonomies and the three other chapters deal with ontologies.

Acknowledgments

The editors would like to thank the chapter authors for their insights and contributions to this book.

The editors would also like to acknowledge the members of the review committee and the associated referees for their involvement in the review process of the book. Their in depth reviewing, criticisms and constructive remarks have significantly contributed to the high quality of the slected papers.

Finally, we thank Springer and the publishing team, and especially T. Ditzinger and J. Kacprzyk, for their confidence in our project.

Nantes, Bordeaux, Tours, Lyon
October 2012

Fabrice Guillet, Bruno Pinaud
Gilles Venturini, Djamel Abdelkader Zighed

Review Committee

All published chapters have been reviewed by 2 or 3 referees and at least one non-french speaking referee (2 for most papers).

- Tomas Aluja (UPC, Spain)
- Nadir Belkhiter (Univ. of Laval, Canada)
- Sadok Ben Yahia (Univ. of Tunis, Tunisia)
- Omar Boussaid (Univ. of Lyon 2, France)
- Paula Brito (Univ. of Porto, Portugal)
- Francisco de A. T. De Carvalho (Univ. Federal de Pernambuco, Brazil)
- Gilles Falquet (Univ. of Geneva, Switzerland)
- Carlos Ferreira (LIAAD INESC Porto LA, Portugal)
- Jean-Gabriel Ganascia (Univ. of Paris 6, France)
- Joao Gama (Univ. of Porto, Portugal)
- Fabien Gandon (INRIA, France)
- Robert Hilderman (Univ. of Regina, Canada)
- Philippe Lenca (Telecom Bretagne, France)
- Henri Nicolas (Univ. of Bordeaux, France)
- Monique Noirhomme (FUNDP, Belgium)
- Jian Pei (Simon Fraser Univ., Canada)
- Pascal Poncelet (Univ. of Montpellier, France)
- Zbigniew Ras (Univ. of North Carolina)
- Jan Rauch (Univ. of Prague, Czech Republic)
- Chiara Renso (KDDLAB — ISTI CNR, Italy)
- Lorenza Saitta (Univ. of Torino, Italy)
- Florence Sédes (Univ. of Toulouse 3, France)
- Dan Simovici (Univ. of Massachusetts Boston, USA)
- Ansaf Salleb-Aouissi (Columbia Univ., USA)
- Yannick Toussaint (Univ. of Nancy, France)
- Stefan Trausan-Matu (Univ. of Bucharest, Romania)
- Rosanna Verde (Univ. of Naples 2, Italy)
- Christel Vrain (Univ. of Orléans, France)
- Jef Wijsen (Univ. of Mons-Hainaut, Belgium)
- Michaël Mcguffin (Ecole de Technologie Supèrieure, Canada)

Associated Reviewers

Hanane Azzag, Nahla Benamor, Julien Blanchard, Marc Boullé, Sylvie Guillaume, Pascale Kuntz, Patrick Marcel, Mathieu Roche, Julien Velcin, Nicolas Voisine.

Contents

Part I Data Mining, Classification and Queries

Part II Ontology and Semantic

List of Contributors

Nathalie Abadie is a geographical and cartographical state works engineer, and a PhD candidate at IGN. She is working on specifications formalisations of geographic databases for their integration. She also works on the implementation and evaluation of the proposed models on the IGN databases.

Rafik Abdesselam is Professor of statistics and data analysis at the University of Lyon. His research and teaching interests include supervised classification, topological learning and methods for data mining. He is also member of various national program committees.

Julien Aligon is PhD student at the Computer Science Laboratory of Université François-Rabelais Tours, France. His research interests include On-Line Analytical Processing, data-warehouses, query personalization and recommendation in databases.

Hanane Azzag is currently associate professor at the University of Paris 13 (France) and a member of machine learning team A3 in LIPN Laboratory. Her main research is in biomimetic algorithms, machine learning and visual data mining. Graduated from USTHB University where she received his engineer diploma in 2001. Thereafter, in 2002 she gained an MSC (DEA) in Artificial Intelligence from Tours University. In 2005, after three years in Tours, she received her PhD degree in Computer Science from the University of Tours.

Giuseppe Berio is professor of computer science at the University of South Brittany, France, and affiliated to the Lab-STICC laboratory. Previously, he was senior researcher at the University of Turin, Italy, working in the Department of Computer Science. Prof. Berio holds the “Laurea” degree (1990) in Computer Science cum Laude from University of Turin, Master (1991) and PhD (1995) degrees both in Information Systems from Polytechnic of Turin. In 1997, he was granted the “Marie Curie Fellowship” from European Communities for a two year assignment to Laboratory for Industrial Engineering and Mechanical Production (LGIPM) of University of Metz, France. His main research work is in information systems and in interoperability of enterprise software applications. He was in the core members of the UEML Thematic Network and INTEROP Network of Excellence (http://www.inteop-noe.org), both projects funded by the European Commission. Prof. Berio is currently member of IFIP Working Group 8.1 "Design and Evaluation of Information Systems" and IFAC Technical Committee 5.3 "Enterprise Integration and Networking".

Marc Boullé graduated from Ecole Polytechnique (France) in 1987 and Sup Telecom Paris in 1989. He is currently a senior researcher in the data mining research group of Orange Labs. His main research interests include statistical data analysis, data mining, especially data preparation and modelling for large databases. He developed regularized methods for feature preprocessing, feature selection and construction, model averaging of selective naive Bayes classifiers and regressors.

Ahmed Bounekkar is Associate Professor at the University Claude Bernard Lyon 1. He works in the field of data analysis, especially in spatial data analysis and diffusion models of pandemic influenza. He also works on multi-objective problems of optimization.

Sandra Bringay received her Ph.D. in 2006 at the University of Picardie Jules Verne in Medical Informatics. She was then a temporary lecturer (ATER) for the CERIM (Center for Studies and Research in Medical Informatics) at the University of Lille 2. Since 2007, she is a lecturer at the University of Montpellier III and she integrated the LIRMM laboratory in the data mining group (TATOO). She works specifically on data mining techniques dedicated to health data. She takes part of knowledge extraction projects dedicated to health data as well as collaborations with private partners.

Michel Buffa teaches Computer Engineering at the University of Nice Sophia-Antipolis. He conducts his research work on Socio-Semantic Web and more specifically on collaborative Knowledge Platforms. Previously He used to work on Underwater Virtual Reality with Professor Peter Sander. In 1994, He was a visiting scientist at the Robotic Institute of the Carnegie Mellon University in Pittsburgh.

Lionel Chauvin is a postdoctoral research assistant in Computer Science at the University of Nantes (France). He received his Ph.D in Computer Science from the University of Angers in 2010. His Ph.D thesis is about cognitive maps, ontologies and conceptual graphs. His current research are about semantic similarities and ontologies.

Fabien Gandon is a enior Researcher at Inria, Leader of the Wimmics research team in the Inria Research Center of Sophia-Antipolis (France). Fabien has a Ph.D. and HDR in Informatics and Computer Science and is a Graduated Engineer in Applied Mathematics from INSA Rouen. His professional interests include: Semantic Web, Ontologies, Knowledge Engineering and Modelling, Mobility, Privacy, Context-Awareness, Web Services and Multi-Agents Systems. His main domain of application is organizational memories (companies, communities, etc.) and knowledge management in general. His personal objectives are to research and teach in the field of applied computer science and informatics, in an international environment. He also participates to W3C working groups.

Dominique Gay received a PhD degree in Computer Science in 2009 from Université de la Nouvelle-Calédonie (Nouméa, New-Caledonia) and Institut National des Sciences Appliquées (Lyon, France). He is currently a researcher in the data mining research group of Orange Labs. His main research interests are about local pattern mining and its use for classification purpose.

David Genest received his Ph.D in Computer Science from the University of Montpellier in 2000. He joined the LERIA at the University of Angers in 2001 as an assistant professor. His research interests are in the area of knowledge representation and especially graphical models with specific interests in conceptual graphs and cognitive maps.

Toader Gherasim is a PhD student at the University of Nantes and affiliated to the LINA laboratory (Computer Science Laboratory of Nantes Atlantique). Toader's research interests include knowledge engineering and ontology learning and evolution. Toader graduated from the Faculty of Automatic Control and Computers of the Politehnica University of Bucharest in 2007 and received a Master in Data Mining from The Polytechnic School of University of Nantes.

Sylvie Guillaume is an Assistant Professor at the University of Auvergne, France and researcher at the Laboratory of Computer Modeling and Optimization Systems (LIMOS). She hold a Ph.D. in Computer Science in 2000 from the University of Nantes, France. Her research domain is positive and negative association rules and quality measures in data mining. Since 2012, she is also working in data mining applied to specific biological problems and particularly in selection of plant phenolic compounds for the formulation of additives in ruminant feed.

Mounira Harzallah graduated from the Ecole Nationale d'Ingénieurs de Tunis (Tunisia). She received a Ph.D. degree in industrial engineering (2000) from the University of Metz (France). She is currently Assistant Professor at the University of Nantes and affiliated to the LINA laboratory (Computer Science Laboratory of Nantes Atlantique). Her research is in the field of enterprise modeling and knowledge engineering, especially focusing on human competence modeling, enterprise interoperability and similarity measures for ontology building and validation. Dr. Harzallah has been involved in various national and international research projects, among them the INTEROP Network of Excellence (http://www.inteop-noe.org) funded by European Commission.

Pascale Kuntz received the M.S. degree in Applied Mathematics from Paris-Dauphine University and the Ph.D. degree in Applied Mathematics from the Ecole des Hautes Etudes en Sciences Sociales, Paris in 1992. From 1992 to 1998 she was assistant professor in the Artificial Intelligence and Cognitive Science Department at the Ecole Nationale Superieure des Telecommunications de Bretagne. In 1998, she joined the Polytechnic School of Nantes University (France), where she is currently professor of Computer Science in the LINA laboratory (UMR 6241). She was the head of the team “KOD - KnOwledge and Decision” for eight years (2003–2011). She is member of the board of the French Speaking Classification Society. Her research interests include classification, graph mining and graph visualization, and post-mining.

Mustapha Lebbah is currently Associate Professor at the University of Paris 13 and a member of Machine learning Team A3, LIPN. His main researches are centred on machine learning (Self-organizing map, Probabilistic and Statistic, unsupervised learning, cluster analysis. Graduated from USTO University where he received his engineer diploma in 1998. Thereafter, he gained an MSC (DEA) in Artificial Intelligence from the Paris 13 University in 1999. In 2003, after three year in RENAULT R&D, he received his PhD degree in Computer Science from the University of Versaille. He is also member of the IEEE, INNS, SFDS, EGC and AML group.

Aymeric Le Dorze is a Ph.D student of the LERIA at the University of Angers since 2010. His master thesis is about expressing Semantic Web ontologies with Answer Set Programming. His Ph.D thesis is about the use of preferences in order to merge cognitive maps

Israël-César Lerman is an Emeritus Professor from the Rennes 1 University, France and a researcher at the IRISA institute of Rennes in the Data and Knowledge Management Department. His research domain is Data Classification and Data Mining. His most important contribution adresses the problem of probabilistic comparison between complex structures in data analysis and in data mining. He recieved the diploma of “Docteur ès Sciences Mathématiques” in 1971 at the Paris 6 University. He wrote two books in 1970 and 1981 and more than one hundred papers, mostly in French. His second book “Classification et Analyse Ordinale des Données” (Dunod, 1981) is on the site `http://thames.cs.rhul.ac.uk/~bcs/books`.

Freddy Limpens is a doctor in Informatics from Nice — Sophia Antipolis University. He conducted his research at Inria research center on possible ways to bridge Social Web and Semantic Web. He is interested in hacker philosopy and DIY, alternative uses of technology, Art/Science/Philosophy connections, social tagging, and collaborative organisation of shared knowledge.

Stéphane Loiseau is a professor in computer science. After a Ph.d at the university of Paris 11-Orsay, he received his Accreditation to Supervise Research (HDR) in January 1998. He is full professor at the Angers university. His major interests are Knowlegde base validation, visual knowledge model (conceptual graphs, semantic map, . . .) and human-machine interaction.

Patrick Marcel is assistant professor at the Computer Science Department and the Computer Science Laboratory of Université François-Rabelais Tours, France, since

1999. He received is PhD from INSA Lyon in 1998. He has more than 30 publications in referred conferences and journals. He has been reviewer for several conferences. His research interests include database query languages, On-Line Analytical Processing, Knowledge Discovery in Databases, query personalization and recommendation in databases.

Ammar Mechouche obtained an engineering degree in computer science from the Science and Technology University in Algiers, in 2004. In 2005, he obtained a master degree in Artificial Intelligence from the Pierre et Marie Curie University in Paris. After that, he obtained his PhD from the University of Rennes 1 in 2009. During his thesis, he worked on semanic web, ontologies and the semantic annotation of brain MRI images. He then look up a postdoc position at the French Mapping Agency (IGN), where he worked at the Cogit laboratory on geodata discovering and integration using ontologies. He also worked on methodologies for comparing heterogenuous ontologies. In late 2010, he joined the Aix-Marseille University and the LSIS laboratory in Marseille as a research assistant. Since 2011, he works at Thales group as a research engineer.

Sébastien Mustière is a geographical and cartographical state works engineer, and the leader of the Cogit laboratory at IGN. He obtained his PhD in computer science from the Pierre et Marie Curie University in 2001. He carries out his research activities on generalisation at IGN. He also carried out a postdoc position in data processing and GIS at Laval University in Canada.

Elsa Negre received her Ph.D. in CS in 2009 from Université François-Rabelais Tours, France. She is currently an Assistant Professor at Université Paris-Dauphine, France. Her research interests include query recommendation and personalization, data warehousing and social network analysis.

Pascal Poncelet is a Professor and the head of the data mining research group (TATOO) in the LIRMM laboratory. Professor Poncelet has previously worked as lecturer (1993–1994), as associate professor respectively in the Mediterannée University (1994–1999) and Montpellier University (1999–2001), as Professor at the Ecole des Mines d'Alès in France where he was also head of the KDD (Knowledge Discovery for Decision Making) team and co-head of the Computer Science Department (2001–2008). His research interest can be summarized as advanced data analysis techniques for emerging applications. He is currently interested in various techniques of data mining with application in Web Mining and Text Mining. He has published a large number of research papers in refereed journals, conference, and workshops, and been reviewer for some leading academic journals. He was also co-head of the French CNRS Group I3 on Data Mining.

Emeric Prouteau obtained his Master degree from the University of Le Mirail in 2010. He was internship at the Cogit laboratory from april 2010 to september 2010.

Julien Rabatel received his Ph.D. degree in Computer Science from the University of Montpellier II, France, in 2011. He was then a member of the data mining group (TATOO) of the LIRMM Laboratory. He is currently a post-doctoral researcher in the Katholieke Universiteit Leuven, Belgium. His research activities mainly concern pattern mining in structured data such as sequential or graph data, as well as its applications in various health (drug design, chemoinformatics) and industrial (sensor data analysis) contexts.

Djamel Abdelkader Zighed is Professor in computer science at the Lyon 2 University. He is the head of the Human Sciences Institute and he was Director of the ERIC Laboratory (University of Lyon). He is also the coordinator of the Erasmus Mundus Master Program on Data Mining and Knowledge Management (DMKM). He is also member of various international and national program committees.

About the Editors

Fabrice Guillet is a CS professor at Polytech'Nantes, the graduate engineering school of University of Nantes, and a member of the "KnOwledge and Decision" team (COD) of the LINA laboratory. He received a PhD degree in CS in 1995 from the "École Nationale Supérieure des Télécommunications de Bretagne", and his Habilitation (HdR) in 2006 from Nantes university. He is a co-founder of the International French-speaking "Extraction et Gestion des Connaissances (EGC)" society. His research interests include knowledge quality and knowledge visualization in the frameworks of Data Mining and Knowledge Management. He has recently co-edited two refereed books of chapter entitled "Quality Measures in Data Mining" and "Statistical Implicative Analysis — Theory and Applications" published by Springer in 2007 and 2008.

Bruno Pinaud received the PhD degree in Computer Science in 2006 from the University of Nantes. He is currently assistant professor at the University of Bordeaux I in the Computer Science Department since September 2008. His current research interests are visual data mining, graph rewriting systems, graph visualization and experimental evaluation in HCI (Human Computer Interaction). He successfully organized the 2012 edition of the EGC Conference.

Gilles Venturini is a CS Professor at François Rabelais University of Tours (France). His main researches interests concern visual data mining, virtual reality, 3D acquisition, biomimetic algorithms (genetic algorithms, artificial ants). He is at the head of the Fovea research group of the CS Laboratory of the University of Tours. He is co-editor in chief of the French New IT Journal (Revue des Nouvelles Technologies de l'Information) and was recently elected as President of the EGC society.

Djamel Abdelkader Zighed is a CS Professor at the Lyon 2 University. He is the head of the Human Sciences Institute and he was Director of the ERIC Laboratory (University of Lyon). He is also the coordinator of the Erasmus Mundus Master Program on Data Mining and Knowledge Management (DMKM). He is also member of various international and national program committees.

Part I
Data Mining, Classification and Queries

A Bayesian Criterion for Evaluating the Robustness of Classification Rules in Binary Data Sets

Dominique Gay and Marc Boullé

Abstract. Classification rules play an important role in prediction tasks. Their popularity is mainly due to their simple and interpretable form. Classification methods combining classification rules that are interesting (w.r.t. a defined interestingness measure) generally lead to good predictions. However, the performance of rule-based classifiers is strongly dependent on the interestingness measure used (e.g. confidence, growth rate, ...) and on the measure threshold to be set for differentiating interesting from non-interesting rules; threshold setting is a non-trivial problem. Furthermore, it can be easily shown that the mined rules are individually non-robust: an interesting (e.g. frequent and confident) rule mined from the training set could be no more confident in a test phase. In this paper, we suggest a new criterion for the evaluation of the robustness of classification rules in binary labeled data sets. Our criterion arises from a Bayesian approach: we propose an expression of the probability of a rule given the data. The most probable rules are thus the rules that are robust. Our Bayesian criterion is derived from this defined expression and allows us to mark out the robust rules from a given set of rules without parameter tuning.

1 Introduction

Among the main data mining tasks, pattern mining has been extensively studied. Association rules [Agrawal et al., 1993] are one of the most popular patterns. In binary data sets, an association rule is an expression of the form $\pi : X \rightarrow Y$, where X (the body) and Y (the consequent) are subsets of Boolean attributes. Intuitively,

Dominique Gay · Marc Boullé
Orange Labs
TECH/ASAP/PROFiling & data mining
2, avenue Pierre Marzin
F-22307 Lannion Cédex, France
e-mail: firstname.name@orange.com

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 3–21.
DOI: 10.1007/978-3-642-35855-5_1

the rule π means that *"when attributes of X are observed, then attributes of Y are often observed"*. The main interest of a rule pattern is its inductive inference power: from now on, if we observe the attributes of X then we will also probably observe attributes of Y. When Y is a class attribute we talk about classification rules. In this paper, we focus on such rules $X \rightarrow c$ (concluding on a class attribute c). Classification rules seem to be favorable for classification tasks (*if an object is described by attributes of X then it is probably of class c*). Recent advances in rule mining have given rise to many rule-based classification algorithms (see, e.g., pioneering work [Liu et al., 1998] or [Bringmann et al., 2009] for a survey). Existing rule-based methods are known for their interpretable form and also to perform quite well in classification tasks. However, we may point out at least two weaknesses:

The Curse of Parameters. The choice of parameter values is crucial but not trivial. The dilemma is well-known: a high frequency threshold may lead to less rules, but also lesser coverage rate and less discriminating power. A low frequency threshold may lead to a huge amount of rules, among which some rules (with low frequency) may be spurious. The same dilemma stands when thresholding interestingness measures like confidence (i.e. an estimation of the probability $P(c \mid X)$) or growth rate (which highlights the so-called emerging patterns, i.e. those patterns that frequent in a class of the data set and barely infrequent in the rest of the data [Dong and Li, 1999]): indeed, high confidence (or growth rate) threshold values lead to strong (pure) class association rules which may be rare in real-world data or even wrong when combined with a low frequency threshold whereas "low" thresholds generate a lot of rules with limited interest. Thus, finding a trade-off between frequency and interestingness measure values is not trivial.

Instability of Interestingness Measures. Even if subsets of extracted rules have shown to be quite effective for predictions, it can be easily shown that highly confident or emerging rules are not individually robust. In figure 1, we compare the confidence (resp. growth rate) train values with the confidence (resp. growth rate) test values of rules extracted from `UCI breast-w` data set [Frank and Asuncion, 2010]. We observe that confidence and growth rate values of extracted rules are clearly unstable from train to test data. The same observation arises when considering lift values: when $lift \geq 2$, then there is a positive correlation between the body of the rule and the class attribute. However, this correlation is not always confirmed in test phase. Thus, confidence, growth rate and lift do not allow us to determine whether a rule is robust: a "good" rule w.r.t. confidence (or growth rate) in training phase may turn out to be weak in test phase.

In this paper, we suggest a Bayesian criterion which allows us to mark out the extracted rules that are robust. Our approach benefits from the `MODL` framework [Boullé, 2006], provides a parameter-free criterion and does not need any wise thresholding. Notice that this paper is the extended English version of the French paper [Gay and Boullé, 2011] presented at EGC 2011 [Khenchaf and Poncelet, 2011].

The rest of the paper is organized as follows: section 2 briefly recalls some needed definitions and the main concepts of the `MODL` approach. Then, we describe our extension of the `MODL` approach for classification rules and a Bayesian criterion

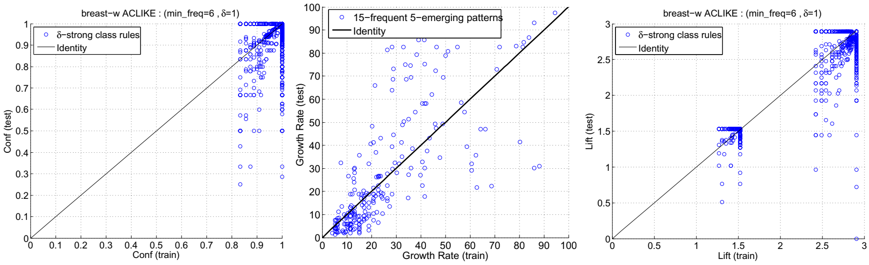


Fig. 1 Comparison of confidence (resp. growth rate and lift) values for classification rules in a train-test experiment: 50% train / 50% test for `breast-w` data set

for evaluating the robustness of rules. Section 3 reports the experiments we led to validate the proposed criterion. We, then discuss further related work in section 4. Finally, section 5 briefly concludes and opens several perspectives for future work.

2 From Classification Rules to MODL Rules

Definitions. Let $r = \{\mathcal{T}, \mathcal{I}, \mathcal{C}, \mathcal{R}\}$ be a binary labeled data set, where $\mathcal{T}$ is a set of objects, $\mathcal{I}$ a set of Boolean attributes, $\mathcal{C}$ a set of classes and $\mathcal{R} : \mathcal{T} \times \mathcal{I} \mapsto \{0,1\}$ a binary relation such that $\mathcal{R}(t,a) = 1$ means object t contains attribute a. Every object $t \in \mathcal{T}$ is labeled by a unique class attribute $c \in \mathcal{C}$. A *classification rule* π in r is an expression of the form $\pi : X \rightarrow c$ where $X \subseteq \mathcal{I}$ is an itemset (i.e., a set of attributes), and $c \in \mathcal{C}$ a class attribute. The *frequency* of itemset (i.e. a set of attributes) X in r is $freq(X,r) = |\{t \in \mathcal{T} \mid \forall a \in X : \mathcal{R}(t,a)\}|$ and the *frequency* of π is $freq(\pi,r) = freq(X \cup \{c\})$. The *confidence* of π in r is $conf(\pi,r) = freq(\pi,r)/freq(X,r)$. The *growth rate* of π is $GR(\pi,r) = freq_r(X,r_c)/freq_r(X,r \setminus r_c)$ where r_c is the data set r restricted to objects of class c ($\mathcal{T}_c$) and $freq_r$ stands for *relative frequency* (i.e. $freq_r(X,r_c) = freq(X,r_c)/|\mathcal{T}_c|$).

The pioneering works in classification based on association rules (i.e. the CBA-like methods, e.g., [Dong et al., 1999, Li et al., 2001, Liu et al., 1998]) state that a rule is interesting for classification if its frequency and confidence (or growth rate) exceed user-defined thresholds. Setting good thresholds may be a hard task for an end-user, therefore low thresholds are arbitrarily set – generating a huge number of rules. Then, a subset of extracted rules is selected in a post-processing phase w.r.t. coverage, redundancy, correlation (e.g. by choosing the k best rules or using the χ^2 test). Therefore, other non-trivial parameter tuning skills are needed.

In this paper, we suggest to follow the MODL approach to evaluate classification rules. The MODL approach, already used for values grouping [Boullé, 2005], discretization [Boullé, 2006], regression [Hue and Boullé, 2007] or decision trees [Voisine et al., 2010], bets on a trade-off between, *(i)* the fineness of the predictive information provided by the model and *(ii)* the robustness, in order to obtain a good generalization of the model. In our context, from a MODL point of view, a model is classification rule. To choose the best rule model, we use a Bayesian approach: we look for maximizing $p(\pi \mid r)$ the posterior probability of a rule model π given the data r. Applying the Bayes theorem and considering the fact that the probability $p(r)$ is constant for a given classification problem, then the expression $p(\pi) \times p(r \mid \pi)$ is to be maximized; where $p(\pi)$ is the prior probability of a rule and $p(r \mid \pi)$, the likelihood, is the conditional probability of the data given the rule model π. Thus, the rule π maximizing this expression, is the most probable rule arising from the data. Our evaluation criterion is based on the negative logarithm of $p(\pi \mid r)$, which we call the *cost* of the rule:

$$c(\pi) = -\log(p(\pi) \times p(r \mid \pi))$$

In order to compute the prior probability $p(Rule)$ of the `MODL` criterion, we propose a definition of a classification rule based on a hierarchy of parameters that uniquely identifies a given rule.

Standard Classification Rule Model. A `MODL` rule (also called *standard classification rule model* (SCRM)) is defined by:

- the constituent attributes of the rule body
- for each attribute of the rule body, the value (0 or 1) that belongs to the body
- the distribution of classes inside and outside of the body

The two last key points of the SCRM definition lead us to a notion of rule that extend the "classical" association and classification rule. Indeed, for a given binary attribute a, the values 0 and 1 are two possible values belonging to the body. This may be related to the notion of rules with negations of attributes in their body (see [Antonie and Zaïane, 2004]). SCRM is also related to the recently introduced *distribution rule* [Jorge et al., 2006]. The consequent of such a rule is a probabilistic distribution over the classes (instead of being a class value). The following example illustrates these two differences.

Example of SCRM. Let us consider the rule $\pi : (A_1 = 0) \wedge (A_2 = 1) \wedge (A_4 = 1) \rightarrow (P_{c_1} = 0.9, P_{c_2} = 0.1)$. Describing the body of such a rule consists in choosing the attributes involved in the body, then choosing the values (0 or 1) of the involved attributes. Notice that a classification rule with negations might be trivially derived from a SCRM using the class with maximum probability as the consequent. For example, $\pi : (A_1 = 0) \wedge (A_2 = 1) \wedge (A_4 = 1) \rightarrow c_1$.

To formally define our evaluation criterion we will use the following additional notations:

Notations. Let r be a binary labeled data set with N objects, m binary attributes and J classes. For a SCRM, $\pi : X \rightarrow (P_{c_1}, P_{c_2}, \ldots, P_{c_J})$ such that $|X| = k \leq m$, we will use the following notations:

- $X = \{x_1, \dots, x_k\}$: the constituent attributes of the rule body ($k \leq m$)
- $i_{x_1}, \dots, i_{x_k}$: the indexes of binary values involved in the rule body
- $N_X = N_{i_{x_1} \dots i_{x_k}}$: the number of objects in the body $i_{x_1} \dots i_{x_k}$
- $N_{\neg X} = N_{\neg i_{x_1} \dots i_{x_k}}$: the number of objects outside of the body $i_{x_1} \dots i_{x_k}$
- $N_{Xj} = N_{i_{x_1} \dots i_{x_k} j}$: the number of objects of class j in the body $i_{x_1} \dots i_{x_k}$
- $N_{\neg Xj} = N_{\neg i_{x_1} \dots i_{x_k} j}$: the number of objects of class j outside of the body $i_{x_1} \dots i_{x_k}$

MODL Hierarchical Prior. We use the following distribution prior on SCRM models, called the MODL hierarchical prior, to define the prior $p(\pi)$.

- *(i)* the number of attributes in the rule body is uniformly distributed between 0 and m
- *(ii)* for a given number k of attributes, every set of k constituent attributes of the rule body is equiprobable
- *(iii)* for a given attribute value, belonging to the body or not are equiprobable
- *(iv)* the distributions of class values in and outside of the body are equiprobable
- *(v)* the distributions of class values in and outside of the body are independent

Thanks to the definition of the model space and its prior distribution, we now apply the Bayes theorem to express the prior probabilities of the model and the probability of the data given the model (i.e. $p(\pi)$ and $p(r \mid \pi)$).
The prior probability $p(\pi)$ of the rule model is:

$$p(\pi) = p(X) \times \prod_{1 \leq l \leq k} p(i_{x_l}) \times \prod_{i \in \{X, \neg X\}} p(\{N_{ij}\} \mid N_X, N_{\neg X})$$

Firstly, we consider $p(X)$ (the probability of having to the attributes of X in the rule body). The first hypothesis of the hierarchical prior is the uniform distribution of the number of constituent attributes between 0 and m. Furthermore, the second hypothesis says that every set of k constituent attributes of the rule body is equiprobable. The number of combinations $\binom{m}{k}$ could be a natural way to compute this prior; however, it is symmetric. Beyond $m/2$, adding new attributes make the selection more probable. Thus, adding irrelevant variables is favored, provided that this has an insignificant impact on the likelihood of the model. As we prefer simpler models, we suggest to use the number of combinations with replacement $\binom{m+k-1}{k}$. Using the two first hypothesis, we have:

$$p(X) = \frac{1}{m+1} \cdot \frac{1}{\binom{m+k-1}{k}}$$

For each attribute x part of the body of the rule, the value involved in the body has to be chosen from $\{0, 1\}$. Thus we have $p(i_x) = 1/2$ (considering hypothesis *(iii)*). Now considering hypothesis *(iv)* and *(v)*, enumerating the distributions of the J classes in and outside of the body turns into a combinatorial problem:

$$p(\{N_{Xj}\} \mid N_X, N_{\neg X}) = \frac{1}{\binom{N_X+J-1}{J-1}}$$

$$p(\{N_{\neg X j}\} \mid N_X, N_{\neg X}) = \frac{1}{\binom{N_{\neg X}+J-1}{J-1}}$$

Concerning the likelihood term, the probability of the data given the model is the probability of observing the data inside and outside of the rule body (with resp. N_X and $N_{\neg X}$ objects) given the multinomial distribution defined for N_X and $N_{\neg X}$. We have:

$$p(r \mid \pi) = \frac{1}{\frac{N_X!}{\Pi_{j=1}^{j=J} N_{X,j}!}} \cdot \frac{1}{\frac{N_{\neg X}!}{\Pi_{j=1}^{j=J} N_{\neg X,j}!}}$$

We now have a complete definition of the cost a MODL rule (SCRM) π:

$$c(\pi) = \log(m+1) + \log\binom{m+k-1}{k} + k\log(2) \tag{1}$$

$$+\log\binom{N_X+J-1}{J-1} + \log\binom{N_{\neg X}+J-1}{J-1} \tag{2}$$

$$+\left(\log N_X! - \sum_{j=1}^{j=J} \log N_{X,j}!\right) + \left(\log N_{\neg X}! - \sum_{j=1}^{j=J} \log N_{\neg X,j}!\right) \tag{3}$$

The cost of the rule is made of negative logarithms of probabilities; according to [Shannon, 1948], this transformation links probabilities with code length. Thus, $c(\pi)$ might be seen as the ability of a MODL rule to encode the classes given the attributes. The first line stands for the choice of the number of attributes, the attributes and the values involved in the rule body. The second line corresponds to the class distribution in and outside of the body. The two last lines stand for the likelihood (the probability of observing the data given the rule).

Intuitively, rules with low MODL cost are the most probable and thus the best ones. Notice that $c(\pi)$ is smaller for lower k values (cf eq. 1), i.e. rules with shorter bodies are more probable thus preferable. Consequently, frequent rules are more probable than non-frequent ones – that meets the obvious fact. From $c(\pi)$ expression again (two last lines), the notion of pureness (fineness) arises: the stronger rules are cheaper w.r.t. c, thus are the best ones. Since the magnitude of the MODL cost of rules depends on the size of the data set (i.e. the number of objects N and the number of attributes m), we define a normalized criterion (noted *level*[1]) to compare two MODL rules:

$$level(\pi) = 1 - \frac{c(\pi)}{c(\pi_\emptyset)}$$

where $c(\pi_\emptyset)$ is the MODL cost for the default rule (i.e. with empty body). Intuitively, $c(\pi_\emptyset)$ is the coding length of the classes when no information is used from the attributes. The cost of the default rule $\pi_\emptyset$ is formally:

$$c(\pi_\emptyset) = \log(m+1) + \log\binom{N+J-1}{J-1} + \log N! - \sum_{j=1}^{j=J} \log N_j!$$

[1] The *level* may also be seen as a compression rate.

That way, for a given rule π, if $level(\pi) = 0$ then π has the same cost as $\pi_\emptyset$; thus π is not more probable than the default rule. When $level(\pi) < 0$, then using π to explain the data is more costly than using the empty rule. In other words, π is less probable than $\pi_\emptyset$ and will not be considered as interesting. The cases where $0 < level(\pi) \leq 1$ highlight the interesting classification rules π. Indeed, rules with lowest cost (and high *level*) are the most probable and show correlations between the rule body and the class attribute. Notice that $level(\pi) = 1$ is the particular case where π (on its own) is sufficient to exactly characterize the class distribution.

We argue that the level allows us to identify the robust and interesting classification rules. In the following, we lead several experiments to support our point of view.

3 Experimentations

In this section, we lead several experiments to show *(i)* that confidence and growth rate are generally unstable from train to test phase and thus are not good candidates to capture the robustness of classification rules; *(ii)* that, conversely, the *level* is stable in the same experimental conditions and *(iii)* that the *level* allows us to naturally identify robust and interesting rules.

3.1 Experimental Protocol

In our experiments, we use seven UCI data sets [Frank and Asuncion, 2010] and a real-world data set (meningite) [François et al., 1992]. A brief description of these data sets is given in table 1.

Table 1 Experimental data sets description

Data set	#Objects	#Attributes	#classes and distribution
breast-w	699	9	458/241
credit-a	690	15	307/383
credit-g	1000	21	700/300
diabetes	768	8	500/268
meningite	329	23	245/84
sonar	208	60	97/111
tic-tac-toe	958	9	626/332
vote	435	17	267/168

The train-test experiments consist in dividing a data set in two (almost) equal class-stratified parts. One part is for training and mining frequent-confident (or emerging) rules, the other part is for evaluating the evolution of confidence and growth rate values on the test set. Since we do not provide an extractor of MODL rules in this preliminary work, we compute the value of our MODL criterion for the extracted confident (or emerging) rules on the training and test set for comparison. We use ACLike prototype [Boulicaut et al., 2003] to mine frequent-confident classification rules: in fact, ACLike mines γ-frequent δ-free itemsets that are bodies of rules π with $conf(\pi, r) \geq 1 - \delta/\gamma$. We also use consepminer prototype [Dong and Li, 1999, Zhang et al., 2000] to mine γ-frequent ρ-emerging patterns.

3.2 Experimental Results

Original Data Sets. In figures 2 and 3, we report scatter plots for the study of the evolution (from train set to test set) of confidence values of extracted rules. We also compare the values of the MODL criterion. As expected, for all data sets, we observe that confidence is unstable from train to test: indeed, a highly confident rule in train may have low confidence in test (see the points far from the identity line). Conversely, the MODL level values of extracted rules are rather stable in the train-test experiments (see the points close to the identity line). A similar experimentation is reported in figures 4, 5 and the same conclusions stand: growth rate values are unstable in a train-test experiment whereas MODL level values of extracted emerging pattern remain stable.

These experiments show that it could be risky to rely on confidence or growth rate values to make predictions since they do not capture the notion of robustness. The stability of the MODL level is a sign of robustness; in the following experiments, we show that patterns with negative level values are non-significant and the ones with positive level values are patterns of interest.

Noisy Data Sets. In order to simulate the presence of class-noise in the breast-w data set, we add uniform noise in the class attribute using the AddNoise function of WEKA [Witten and Frank, 2005] – with various ratio: 20% and 50% amount of noisy class labels. We then proceed the train-test experiments on each artificially noisy data set. For each amount of noise (see in figure 6), classical extractors (frequent-confident rules and emerging patterns miners) succeed in outputing a set of “potentially” interesting patterns – notice that less rules arise from the most noisy contexts. However, once again the train-test experiments show the instability of classical measures. Moreover, the instability is emphasized in noisy contexts; indeed, most of the points (rules) in the scatter plots (and all rules for 50% of noise) are under the identity line, which means confidence and growth rate are wrongly optimistic and may lead to bad predictions. As an example, several rules confidence fall under 0.5 in the test set – which is contradictory.

The *level* criterion of extracted patterns is still stable in noisy contexts. Notice that most of the confident or emerging rules in noisy contexts has a negative level. As we mentioned above, a rule with a negative level is less probable than the default rule and thus is not statistically significant, i.e. not interesting. In the last experiments, we show that a positive level indicates that a rule is interesting.

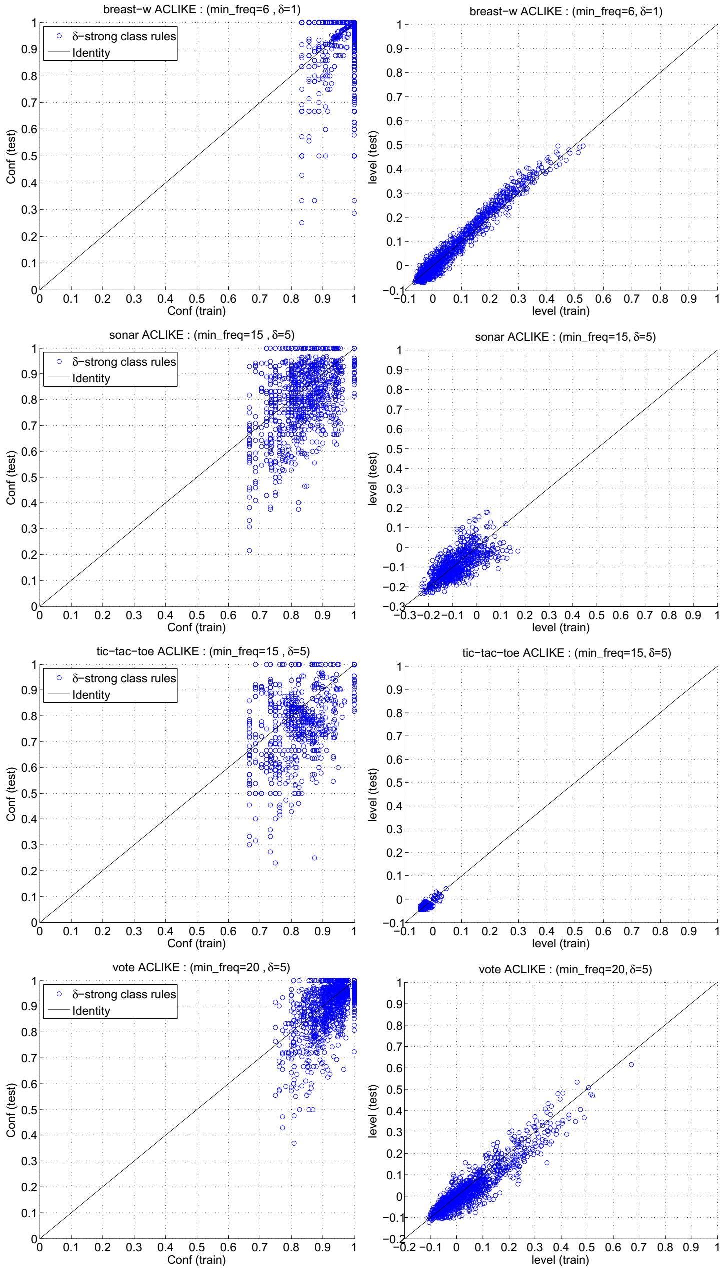


Fig. 2 Comparison of *confidence* and *level*: train values vs test values. Confidence is unstable from train to test phase while *level* values are clearly stable (points close to the identity line) – ensuring the robustness of the criterion.

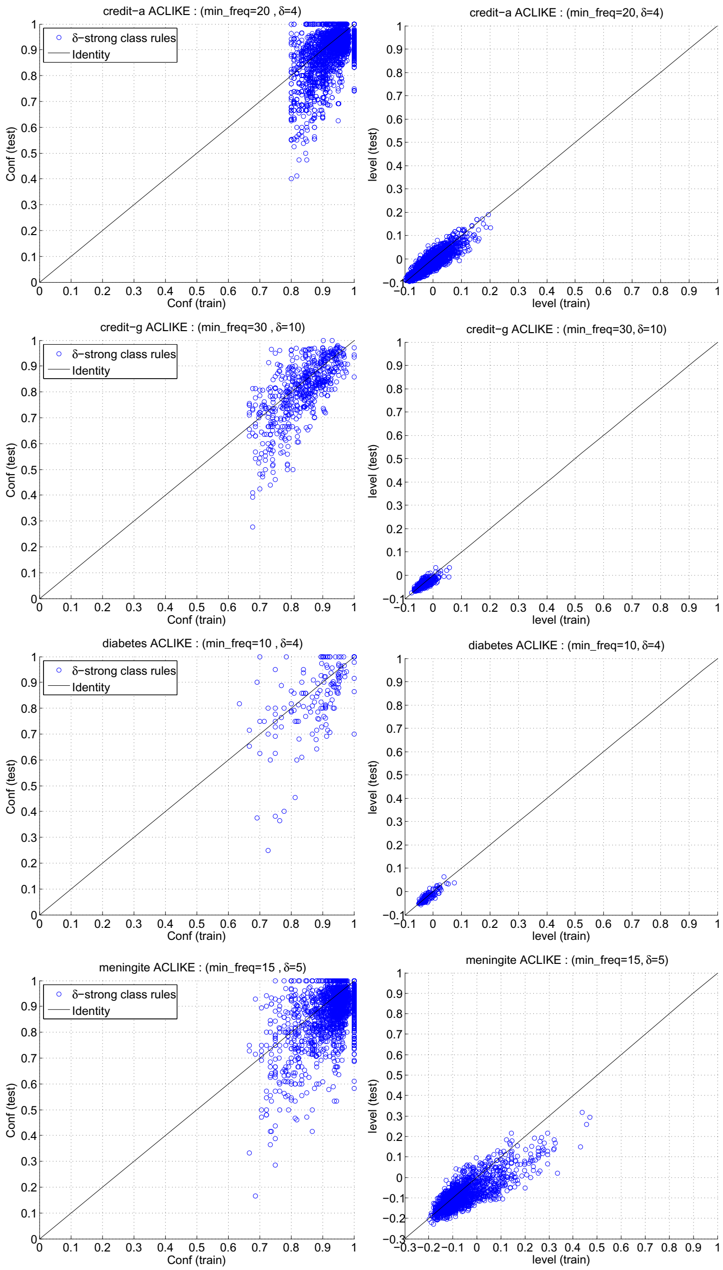


Fig. 3 Comparison of *confidence* and *level*: train values vs test values

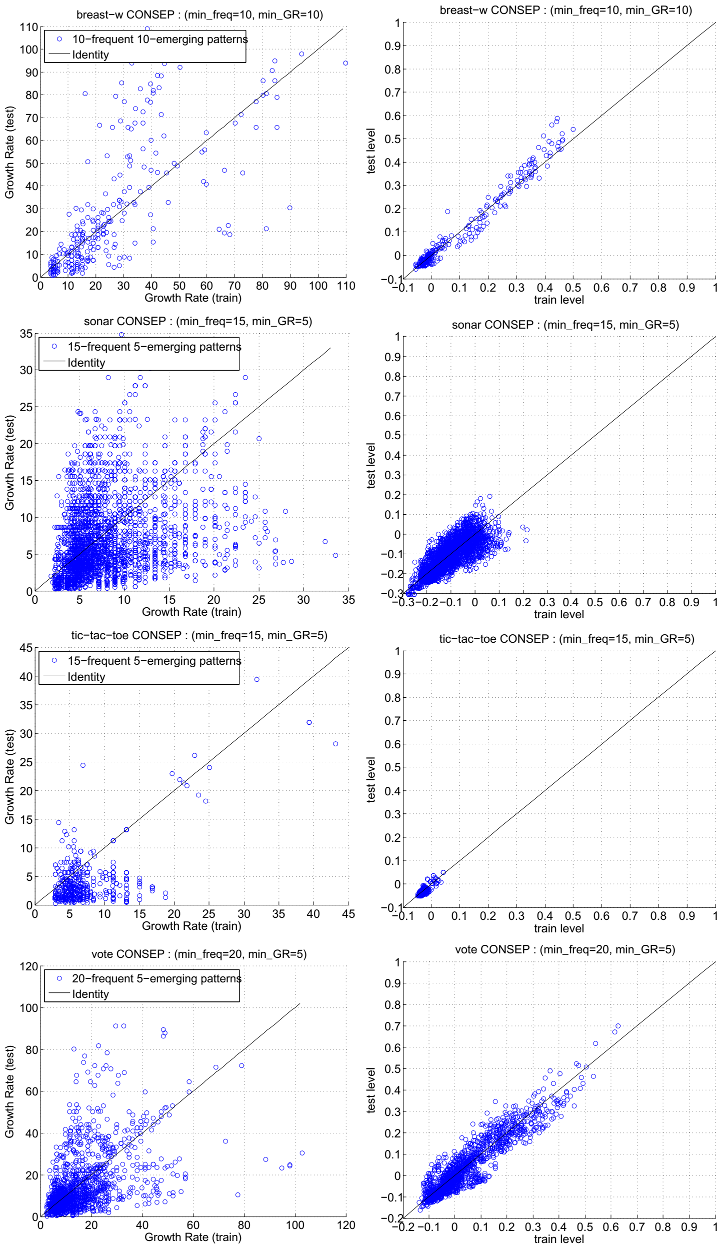


Fig. 4 Comparison of *GR* and *level*: train values vs test values. Growth rate shows instability in train-test experiments while *level* still remains stable.

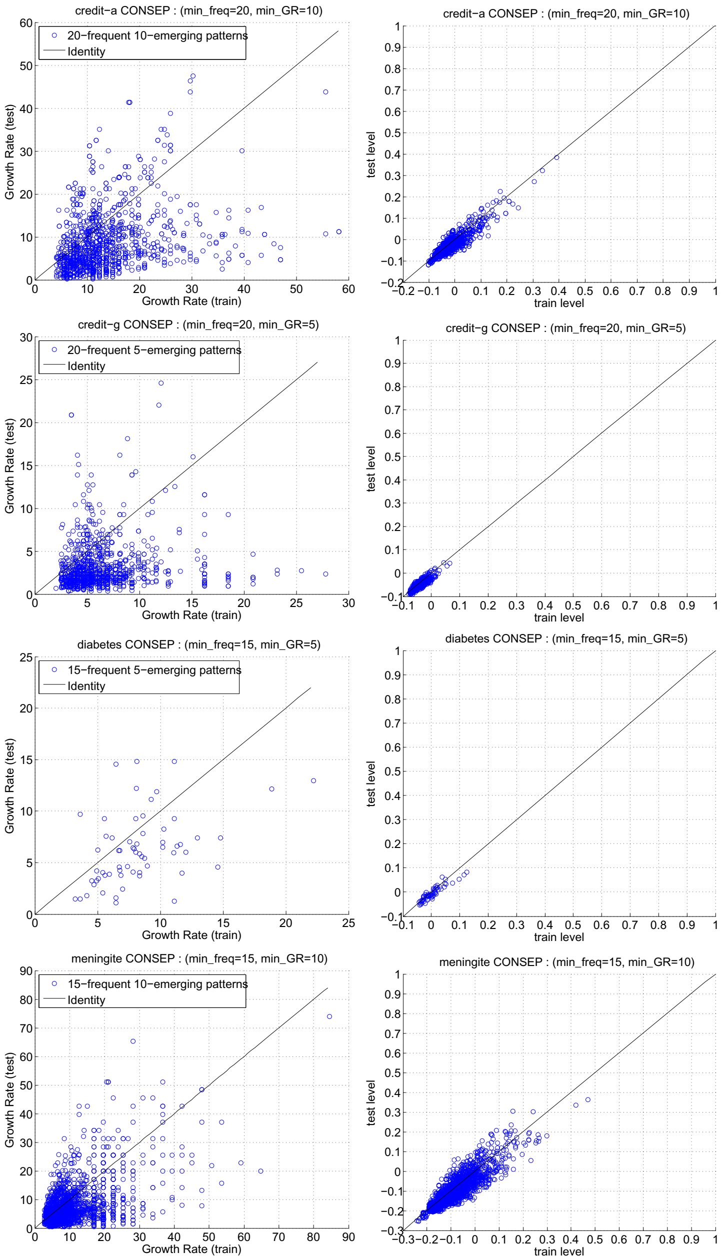


Fig. 5 Comparison of *GR* and *level*: train values vs test values

Patterns with Positive Level. In figures 7 and 8, we report the train and test values of a class-entropy-based measure μ (defined below) for the extracted rules π:

$$\mu(\pi) = N \times (Ent(\pi_\emptyset) - Ent(\pi))$$

μ measures the difference between the conditional entropies of the null rule model (default rule) and a given rule π. The higher μ, the more interesting π is. μ may be seen as the number of bits saved when compressing the data using π instead of using $\pi_\emptyset$. In figures 7 and 8, we highlight the rules with a positive MODL level (red 'o'). As expected, rules with a positive level are generally the most interesting, i.e. with higher μ values. Consequently, rules with a negative level (blue '+') value are located in the southwest of the graphs, with low μ values.

4 Related Work and Discussions

The MODL approach [Boullé, 2005, Boullé, 2006] and the *level* criterion are at the crossroads of Bayes theory, Minimum Description Length principle (MDL [Grünwald, 2007]) and Kolmogorov complexity [Li and Vitányi, 2008].

About MDL. In [Siebes et al., 2006], the authors develop a MDL-based pattern mining approach. The authors look for itemsets that provides a good compression of the data. The link between probability and codes allow them to rewrite the code length of an item set I as $-\log(P(I))$. Thus, the best item sets have shortest codes. In [van Leeuwen et al., 2006], an extension for classification purpose is suggested. The two main differences with the MODL approach are: *(i)* the use of the MODL hierarchical prior implies a different way of coding information; *(ii)* in [van Leeuwen et al., 2006], authors look for a set of patterns to compress the data whereas our MODL criterion is defined for *one* rule.

Notice that another recent work embraces the MDL principle for classification rule discovery: in [Suzuki, 2009], the author suggests an extended version of MDL to integrate user knowledge (in the form of a partial decision list). The code length cl of the partial decision list L to be discovered from data D is extended with the user knowledge K and serves as a subjective interestingness measure: $cl(L) \equiv -\log P(L) - \log P(D \mid L) - \log P(K \mid L)$.

About Robustness. The *level* criterion has shown to be stable. Thus, we may rely on classification rules with positive *level* values since the interestingness of the rules will be confirmed in a test phase. The notion of robustness has been studied recently: in [Le Bras et al., 2010], the authors suggest a new notion of robustness dependent on an interestingness measure μ and a threshold μ_{min}. Starting from the observation that a rule can be characterized by a $\mathbb{R}^3$-vector of three values of its contingency table (e.g. the frequency of the body, the frequency of the target and the number of counterexamples; see figure 2), the authors define the robustness of rule π as the normalized Euclidean distance $rob(\pi, \mu_{min}) = ||\pi - \pi^*||_2/\sqrt{3}$ between π and a

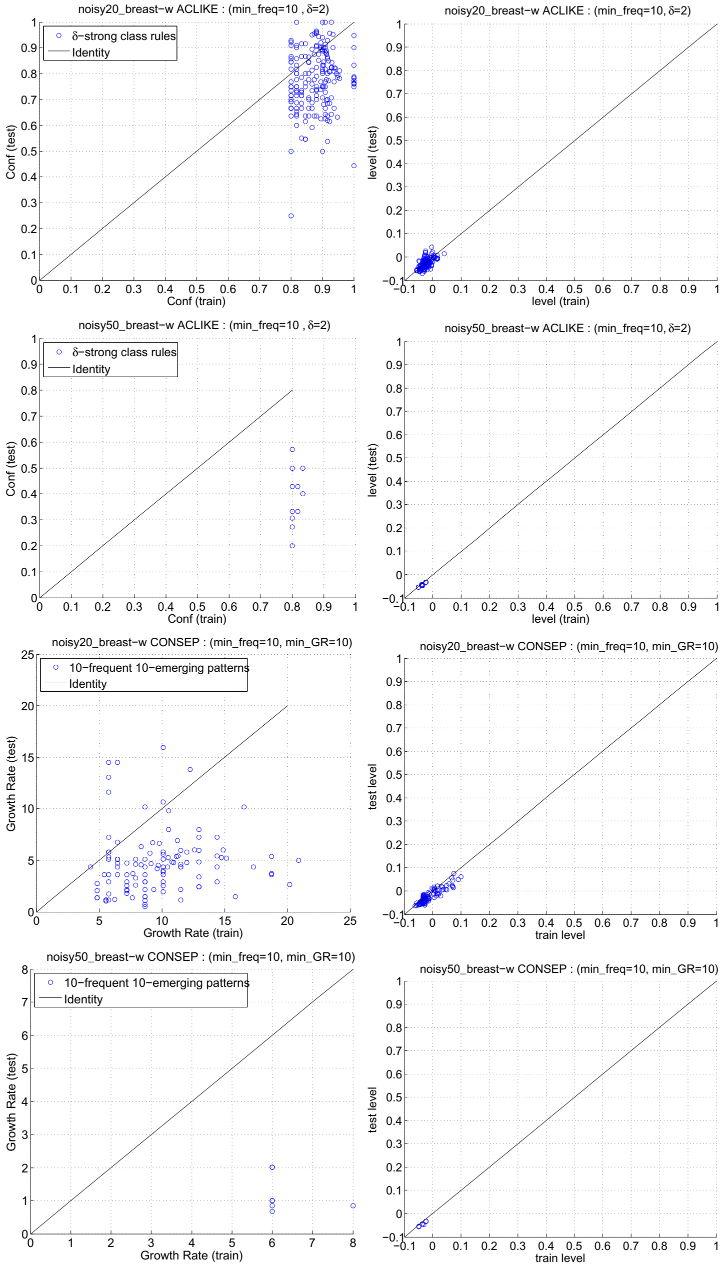


Fig. 6 Comparison *GR*, *confidence* and *level* in artificially noisy `breast-w` data set: train values vs test values. Potentially interesting rules w.r.t. confidence (or growth rate), that are actually 'wrong' in highly noisy environment, have a negative *level* value.

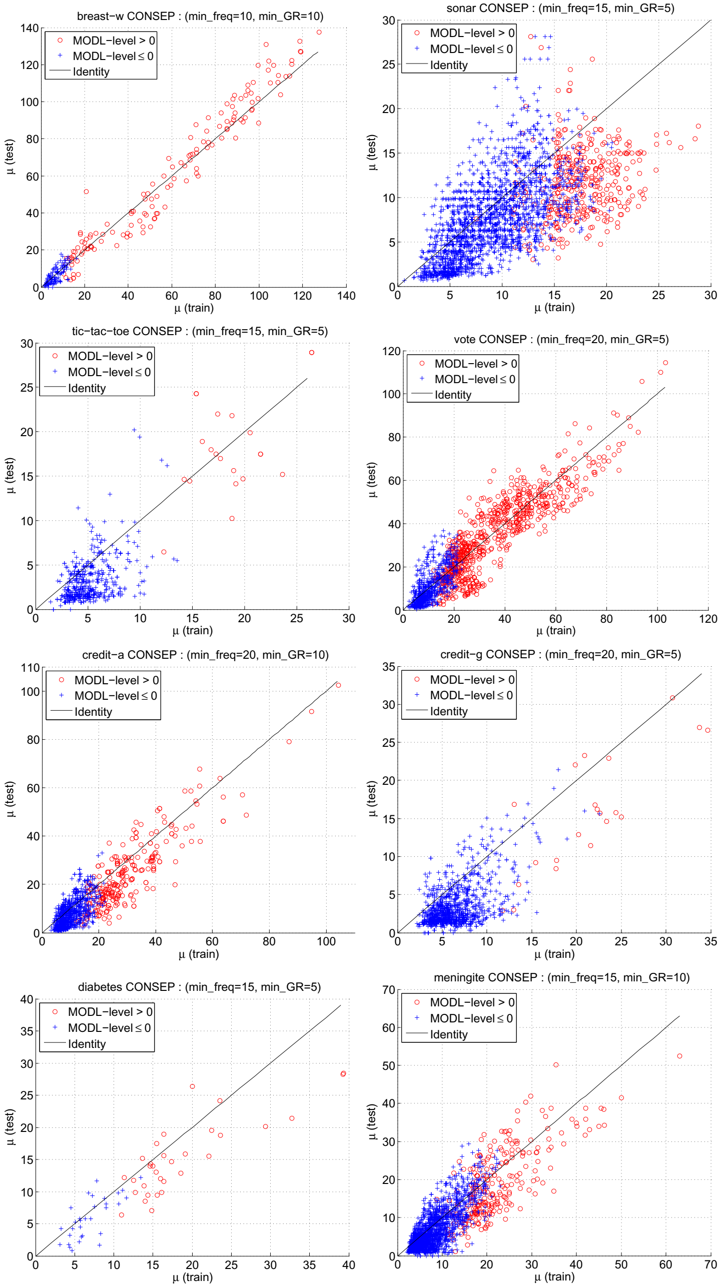


Fig. 7 Comparison μ: train values vs test values of emerging rules. The best rules (i.e. the most probable ones with a positive *level* value, red ‘o’) are generally located at the north-east of the graph whereas non-robust one (with negative *level* value, blue ‘+’) are close to the origin.

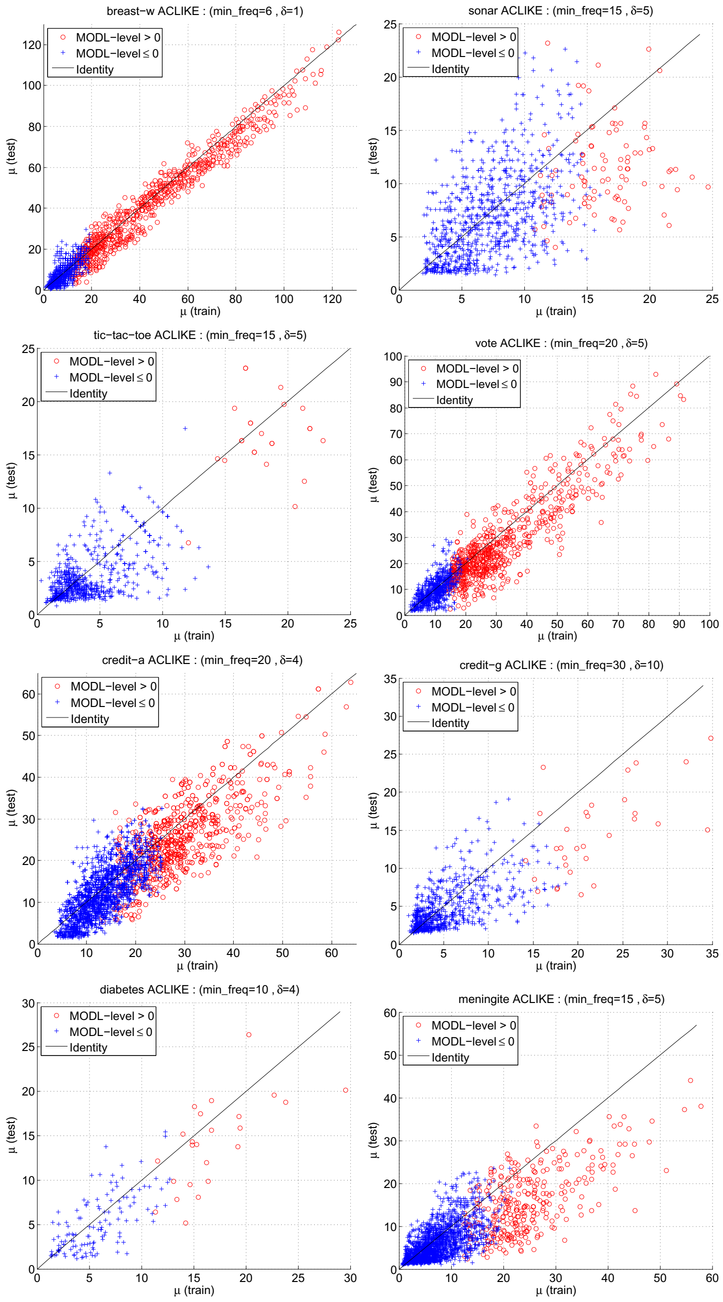


Fig. 8 Comparison μ: train values vs test values of confident rules

Table 2 Contingency table for a classification rule $X \rightarrow c$

$X \rightarrow c$	c	$\neg c$	Σ
X	$freq(Xc,r)$	$freq(X\neg c,r)$	$freq(X,r)$
$\neg X$	$freq(\neg Xc,r)$	$freq(\neg X\neg c,r)$	$freq(\neg X,r)$
Σ	$\|c\|$	$\|\neg c\|$	N

limit rule π^* (i.e. a rule minimizing $g(\pi') = ||\pi' - \pi_{min}||_2$ where π_{min} is such that $\mu(\pi_{min}) = \mu_{min}$). In such framework, comparing two rules in terms of robustness does not need any thresholding, however for filtering purpose (e.g., selection of a subset of robust rules) another non-trivial parameter (*rob*) has to be set (in addition with frequency and the current measure thresholds).

About Redundancy. A classification rule $\pi_2 : Y \rightarrow c_i$ is said to be redundant w.r.t. $\pi_1 : X \rightarrow c_j$ if $c_i = c_j$, $X \subseteq Y$ and π_1 and π_2 brings (almost) the same class-discriminating power (w.r.t. an interestingness measure) – a redundant rule should be pruned. Consider two itemsets X and Y such that $X \subseteq Y$ and $freq(X,r) = freq(Y,r)$, then for a given interestingness measure m based on frequency, we have $m(X) = m(Y)$ thus some redundancy. It is common to consider support equivalence class to group itemsets having the same support (and frequency). The unique longest itemset (w.r.t. set inclusion) is the closed itemset [Pasquier et al., 1999] and the smallest ones are called the free itemsets [Boulicaut et al., 2003]. In state-of-the-art pattern-based methods for classification purpose, the intuition tells that free itemsets should be preferred [Baralis and Chiusano, 2004]. This intuition is confirmed by our *level* criterion. Indeed, if Y is a closed itemset and X a free itemset from the same support equivalence class, then $c(\pi_2 : Y \rightarrow c_i) \geq c(\pi_1 : Y \rightarrow c_i)$ since the number of attributes favors π_1 (line 1–2); and π_1 should be preferred. The main idea is translated in the following proposition (the proof is almost direct when one observes that only the terms of the cost expression that involve parameter k imply a difference of *level* between X and Y):

Proposition 1. *Let X and Y be two itemsets such that $X \subset Y$ and $freq(X,r) = freq(Y,r)$. X is preferable to Y according to the level criterion; i.e., $level(X) > level(Y)$.*

5 Conclusion and Perspectives

In this paper, we have presented a new Bayesian criterion for the evaluation of classification rules in binary data sets. Based on the `MODL` approach (and the `MDL` principle), the new criterion overcomes two well-known drawbacks of existing approaches (using a frequency-confidence or growth rate framework): the non-trivial tuning of interestingness measure threshold and the non-stability of interestingness measure values from train to test phase. Our new criterion, the `MODL` level, promotes a trade-off between fineness and reliability and allows us to easily distinguish interesting rules (with a positive level value) from non-significant rules (with a negative level value) without parameter tuning. Furthermore, the criterion is shown to be robust and gives a true idea of the prediction power of extracted patterns. The experiments we led on UCI data sets confirm both the relevancy and robustness of the criterion. In this preliminary work, we use the MODL criterion in a post-processing step to select interesting and robust rules from a large set of confident or emerging rules. The next step is a constructive approach for mining classification rules with positive MODL level values. Since the MODL approach is also suitable for continuous and nominal attributes as well, another step will be the extension towards quantitative association rules by considering discretization and values grouping.

References

[Agrawal et al., 1993] Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Proceedings ACM SIGMOD 1993, pp. 207–216 (1993)

[Antonie and Zaïane, 2004] Antonie, M.-L., Zaïane, O.R.: An associative classifier based on positive and negative rules. In: DMKD 2004 (2004)

[Baralis and Chiusano, 2004] Baralis, E., Chiusano, S.: Essential classification rule sets. ACM Transactions on Database Systems 29(4), 635–674 (2004)

[Boulicaut et al., 2003] Boulicaut, J.-F., Bykowski, A., Rigotti, C.: Free-sets : A condensed representation of boolean data for the approximation of frequency queries. Data Mining and Knowledge Discovery 7(1), 5–22 (2003)

[Boullé, 2005] Boullé, M.: A bayes optimal approach for partitioning the values of categorical attributes. Journal of Machine Learning Research 6, 1431–1452 (2005)

[Boullé, 2006] Boullé, M.: MODL: A bayes optimal discretization method for continuous attributes. Machine Learning 65(1), 131–165 (2006)

[Bringmann et al., 2009] Bringmann, B., Nijssen, S., Zimmermann, A.: Pattern-based classification: A unifying perspective. In: LeGo 2009 Workshop co-located with EMCL/PKDD 2009 (2009)

[Dong and Li, 1999] Dong, G., Li, J.: Efficient mining of emerging patterns: discovering trends and differences. In: Proceedings KDD 1999, pp. 43–52. ACM Press (1999)

[Dong et al., 1999] Dong, G., Zhang, X., Wong, L., Li, J.: CAEP: Classification by Aggregating Emerging Patterns. In: Arikawa, S., Nakata, I. (eds.) DS 1999. LNCS (LNAI), vol. 1721, pp. 30–42. Springer, Heidelberg (1999)

[François et al., 1992] François, P., Crémilleux, B., Robert, C., Demongeot, J.: MENINGE: a medical consulting system for child's meningitis study on a series of consecutive cases. Artificial Intelligence in Medecine 4(4), 281–292 (1992)

[Frank and Asuncion, 2010] Frank, A., Asuncion, A.: UCI machine learning repository (2010),
http://archive.ics.uci.edu/ml

[Gay and Boullé, 2011] Gay, D., Boullé, M.: Un critère bayésien pour évaluer la robustesse des règles de classification. In: EGC 2011. Revue des Nouvelles Technologies de l'Information, vol. RNTI-E-20, pp. 539–550. Hermann-Éditions (2011)

[Grünwald, 2007] Grünwald, P.: The minimum description length principle. MIT Press (2007)

[Hue and Boullé, 2007] Hue, C., Boullé, M.: A new probabilistic approach in rank regression with optimal bayesian partitioning. Journal of Machine Learning Research 8, 2727–2754 (2007)

[Jorge et al., 2006] Jorge, A.M., Azevedo, P.J., Pereira, F.: Distribution Rules with Numeric Attributes of Interest. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) PKDD 2006. LNCS (LNAI), vol. 4213, pp. 247–258. Springer, Heidelberg (2006)

[Khenchaf and Poncelet, 2011] Khenchaf, A., Poncelet, P. (eds.): Extraction et gestion des connaissances (EGC 2011), Janvier 25-29, Brest, France. Revue des Nouvelles Technologies de l'Information, vol. RNTI-E-20. Hermann-Éditions (2011)

[Le Bras et al., 2010] Le Bras, Y., Meyer, P., Lenca, P., Lallich, S.: A Robustness Measure of Association Rules. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010, Part II. LNCS (LNAI), vol. 6322, pp. 227–242. Springer, Heidelberg (2010)

[Li and Vitányi, 2008] Li, M., Vitányi, P.M.B.: An Introduction to Kolmogorov Complexity and Its Applications, 3rd edn. Springer (2008)

[Li et al., 2001] Li, W., Han, J., Pei, J.: CMAR: Accurate and efficient classification based on multiple class-association rules. In: Proceedings ICDM 2001, pp. 369–376. IEEE Computer Society (2001)

[Liu et al., 1998] Liu, B., Hsu, W., Ma, Y.: Integrating classification and association rule mining. In: Proceedings KDD 1998, pp. 80–86. AAAI Press (1998)

[Pasquier et al., 1999] Pasquier, N., Bastide, Y., Taouil, R., Lakhal, L.: Efficient mining of association rules using closed itemset lattices. Information Systems 24(1), 25–46 (1999)

[Shannon, 1948] Shannon, C.E.: A mathematical theory of communication. Bell System Technical Journal (1948)

[Siebes et al., 2006] Siebes, A., Vreeken, J., van Leeuwen, M.: Item sets that compress. In: SIAM DM 2006 (2006)

[Suzuki, 2009] Suzuki, E.: Negative Encoding Length as a Subjective Interestingness Measure for Groups of Rules. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 220–231. Springer, Heidelberg (2009)

[van Leeuwen et al., 2006] van Leeuwen, M., Vreeken, J., Siebes, A.: Compression Picks Item Sets That Matter. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) PKDD 2006. LNCS (LNAI), vol. 4213, pp. 585–592. Springer, Heidelberg (2006)

[Voisine et al., 2010] Voisine, N., Boullé, M., Hue, C.: A Bayes Evaluation Criterion for Decision Trees. In: Guillet, F., Ritschard, G., Zighed, D.A., Briand, H. (eds.) Advances in Knowledge Discovery and Management. SCI, vol. 292, pp. 21–38. Springer, Heidelberg (2010)

[Witten and Frank, 2005] Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann (2005)

[Zhang et al., 2000] Zhang, X., Dong, G., Ramamohanarao, K.: Exploring constraints to efficiently mine emerging patterns from large high-dimensional datasets. In: KDD 2000, pp. 310–314 (2000)

Mining Sequential Patterns: A Context-Aware Approach

Julien Rabatel, Sandra Bringay, and Pascal Poncelet

Abstract. Traditional sequential patterns do not take into account contextual information associated with sequential data. For instance, when studying purchases of customers in a shop, a sequential pattern could be "*frequently, customers buy products A and B at the same time, and then buy product C*". Such a pattern does not consider the age, the gender or the socio-professional category of customers. However, by taking into account contextual information, a decision expert can adapt his/her strategy according to the type of customers. In this paper, we focus on the analysis of a given context (e.g., a category of customers) by extracting context-dependent sequential patterns within this context. For instance, given the context corresponding to young customers, we propose to mine patterns of the form "*buying products A and B then product C is a general behavior in this population*" or "*buying products B and D is frequent for young customers only*". We formally define such context-dependent sequential patterns and highlight relevant properties that lead to an efficient extraction algorithm. We conduct our experimental evaluation on real-world data and demonstrate performance issues.

Julien Rabatel
Tecnalia, Cap Omega, Rd-Pt B. Franklin, 34960 Montpellier Cedex 2, France,
LIRMM (CNRS UMR 5506), Univ. Montpellier 2, 161 rue Ada,
34095 Montpellier Cedex 5, France
e-mail: rabatel@lirmm.fr

Sandra Bringay
LIRMM (CNRS UMR 5506), Univ. Montpellier 2, 161 rue Ada,
34095 Montpellier Cedex 5, France,
Dpt. MIAP, Univ. Montpellier 3, Route de Mende, 34199 Montpellier Cedex 5, France
e-mail: bringay@lirmm.fr

Pascal Poncelet
LIRMM (CNRS UMR 5506), Univ. Montpellier 2, 161 rue Ada,
34095 Montpellier Cedex 5, France
e-mail: poncelet@lirmm.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 23–41.
DOI: 10.1007/978-3-642-35855-5_2

1 Introduction

Sequential pattern mining is an important problem widely addressed by the data mining community, with a very large field of applications including the analysis of user behavior, sensor data, DNA arrays, clickstreams, etc. Sequential pattern mining aims at extracting sets of items commonly associated over time. For instance, when studying purchases of customers in a supermarket, a sequential pattern could be *"many customers buy products A and B, then buy product C"*. However, data are very often provided with additional information about purchases, such as the age or the gender of customers. Traditional sequential patterns do not take into account this information. Having a better knowledge about the features of objects supporting a given behavior can help decision making. In this paper, the set of such descriptive information about objects is referred to as contextual information. In the supermarket scenario, the decision expert can adapt his/her strategy by considering the fact that a pattern depends on the type of customer. For instance, an expert who wants to study in detail the *young* customers population could be interested in questions such as "*Are there buying patterns that are frequent for young people, whatever their gender?*" or "*Is there a certain behavior frequent for young people exclusively?*".

Nevertheless, mining such context-dependent sequential patterns is a difficult task. Different contexts should be mined independently to test whether frequent patterns are shared with other contexts. Moreover, some contexts can be more or less general. For instance, the context corresponding to *young* customers is more general than the context corresponding to *young male* customers. Hence, a large number of contexts (more or less general) have to be considered, and the mining process can be very time consuming.

Related Work

Some work in the literature can be seen as related to context-dependent sequential pattern mining. For instance, [Hilderman et al., 1998] define characterized itemsets, i.e., frequent itemsets extracted from a transaction database and associated with values on external attributes defined over a concept hierarchy. Such values can then be generalized using attribute-oriented generalization. Although this notion handles itemsets only, the definitions related to the characterization of patterns could be directly adapted to sequential patterns. However, this work does not take into consideration the representativity of a frequent itemset in a context by considering whether the minimum frequency constraint is satisfied in the sub-contexts. Indeed, an itemset can be associated with a context even if it is not frequent in one or more of its sub-contexts.

Let us consider also the problem of mining multidimensional sequential patterns, i.e., extracting sequential patterns dealing with several dimensions. The first proposition is described in [Pinto et al., 2001], where sequential patterns are extracted in data similar to the contextual data considered in our approach.

However, while multidimensional sequential patterns consider contextual information, they associate a context to a sequential pattern only if this association is globally frequent in the whole database. Moreover, similarly to characterized itemsets, multidimensional sequential patterns do not consider their representativity in their corresponding context. As a consequence, a sequential pattern which is not frequent in the whole database can not be extracted, even if it is very frequent in a sub-category of customers (e.g., the old female customers). We claim in this paper that such a pattern can however be very interesting for a decision maker. Other approaches have considered more complex multidimensional sequence databases, e.g., where items are also described over several dimensions [Plantevit et al., 2005], but the same principles is used to extract such patterns, then leading to the same problems. The same remark can also be mentioned in [Ziembiński, 2007].

To the best of our knowledge, the first approach that tackles this problem for sequential patterns being dependent on more or less general contexts has been proposed in [Rabatel et al., 2010]. However, this work focuses on mining sequential patterns in the whole hierarchy of contexts. Here, we are interested in a different application case: the user focuses on a given context (e.g., young customers) and aims at studying behaviors being related to this context only. We show in this paper that this approach exhibits some interesting properties that can be used in order to efficiently mine context-dependent sequential patterns. In addition, we propose a new type of context-dependent sequential patterns: the exclusive sequential patterns. Intuitively, exclusive sequential patterns are frequent and representative within a context and do not appear frequently elsewhere.

The problem of mining emerging patterns can also be seen as related to context-dependent patterns. Introduced in [Dong and Li, 1999], the mining of emerging patterns aims to extract the itemsets that are discriminant to one class in a set of itemset databases. An emerging pattern is then a pattern whose support is significantly higher in a class than in others. Such patterns can then be exploited to build classifiers [Dong et al., 1999, Li et al., 2001]. For instance, given two data classes *A* and *B*, emerging patterns in *B* would be itemsets whose support is significantly higher in *B* than in *A*. Globally, although emerging patterns and context-dependent sequential patterns (in particular, exclusive sequential patterns) aim at mining patterns that are more frequent in one database than in others, there are some important differences. In particular, the most important problem when mining context-dependent patterns is related to the generalization / specialization order existing amongst contexts. For instance, the context corresponding to young people is more general than the one corresponding to young male people. This aspect is not considered in the mining of emerging patterns, where classes of data do not have such an ordering relation. In addition, our work is only based on the frequency of patterns, while emerging patterns consider a ratio of frequencies over several classes of data.

Contributions

In this paper, we first formally describe contexts. Then, by highlighting relevant properties of such contexts, we show how sequential patterns dependent on one context can be extracted. We conduct experimental evaluation on real-world data and demonstrate performance issues.

More precisely, the following is organized as follows. In Section 2, we define the "traditional" sequential pattern mining problem and show why it is not relevant when contextual information is available. Contextual data, as well as context-dependent sequential patterns, are presented in Section 3. In Section 4, we highlight some relevant properties of context-dependent sequential patterns that are exploited to propose an efficient algorithm. In Section 5, conducted experiments on a real-world dataset are presented. We conclude and discuss future work in Section 6.

2 Problem Definition

2.1 Traditional Sequential Patterns

This section describes the traditional sequential pattern mining problem and highlights the need for a specific way to handle contextual information.

Sequential patterns were introduced in [Agrawal and Srikant, 1995] and can be considered as an extension of the concept of frequent itemset [Agrawal et al., 1993] by handling timestamps associated to items. Sequential pattern mining aims at extracting sets of items commonly associated over time. In the *"basket market"* scenario, a sequential pattern could be: *"40 % of the customers buy a television, then buy later a DVD player"*. The problem of mining all sequential patterns in a sequence database is defined as follows.

Let $\mathcal{X}$ be a set of distinct ***items***. An ***itemset*** is a subset of items, denoted by $I = (i_1 i_2 \ldots i_n)$, i.e., for $1 \leq j \leq n$, $i_j \in \mathcal{X}$. A ***sequence*** is an ordered list of itemsets, denoted by $\langle I_1 I_2 \ldots I_k \rangle$, where $I_i \subseteq \mathcal{X}$ for $1 \leq i \leq n$.

Let $s = \langle I_1 I_2 \ldots I_m \rangle$ and $s' = \langle I'_1 I'_2 \ldots I'_n \rangle$ two sequences. The sequence s is a ***subsequence*** of s', denoted by $s \sqsubseteq s'$, if $\exists i_1, i_2, \ldots i_m$ with $1 \leq i_1 < i_2 < \ldots < i_m \leq n$ such that $I_1 \subseteq I'_{i_1}$, $I_2 \subseteq I'_{i_2}$, $\ldots$, $I_m \subseteq I'_{i_m}$. If $s \sqsubseteq s'$ we also say that s' ***supports*** s.

A ***sequence database*** $\mathcal{D}$ is a relation $\mathcal{R}(ID, S)$, where an element $id \in dom(ID)$ is a sequence identifier, and $dom(S)$ is a set of sequences. The ***size*** of $\mathcal{D}$, denoted by $|\mathcal{D}|$, is the number of tuples in $\mathcal{D}$. A tuple $\prec id, s \succ$ is said to *support* a sequence α if α is a subsequence of s, i.e., $\alpha \sqsubseteq s$. The ***support*** of a sequence α in the sequence database $\mathcal{D}$ is the number of tuples in $\mathcal{D}$ supporting α, i.e., $sup_{\mathcal{D}}(\alpha) = |\{\prec id, s \succ | (\prec id, s \succ \in \mathcal{D}) \wedge (\alpha \sqsubseteq s)\}|$.

Given a real $minSup$ such that $0 < minSup \leq 1$ as the ***minimum support threshold***, a sequence α is ***frequent*** in the sequence database $\mathcal{D}$ if the proportion of tuples in $\mathcal{D}$ supporting α is greater than or equal to $minSup$, i.e., $sup_{\mathcal{D}}(\alpha) \geq minSup \times |\mathcal{D}|$. In this case, sequence α is also called a ***sequential pattern*** in $\mathcal{D}$.

Example 1. *Table 1 shows a sequence database describing the purchases of customers in a shop. The first column stands for the identifier of each sequence given in the last column. a, b, c, d, e are the products. Column Gender and Age represent*

Table 1 A contextual sequence database

id	Age	Gender	Sequence
s_1	young	male	$\langle(ad)(b)\rangle$
s_2	young	male	$\langle(ab)(b)\rangle$
s_3	young	male	$\langle(a)(a)(b)\rangle$
s_4	young	male	$\langle(c)(a)(bc)\rangle$
s_5	young	male	$\langle(d)(ab)(bcd)\rangle$
s_6	young	female	$\langle(b)(a)\rangle$
s_7	young	female	$\langle(a)(b)(a)\rangle$
s_8	young	female	$\langle(d)(a)(bc)\rangle$
s_9	old	male	$\langle(ab)(a)(bd)\rangle$
s_{10}	old	male	$\langle(bcd)\rangle$
s_{11}	old	male	$\langle(bd)(a)\rangle$
s_{12}	old	female	$\langle(e)(bcd)(a)\rangle$
s_{13}	old	female	$\langle(bde)\rangle$
s_{14}	old	female	$\langle(b)(a)(e)\rangle$

extra information about sequences. Such information is not considered in traditional sequential pattern mining. The size of $\mathcal{D}$ is $|\mathcal{D}| = 14$.

The first sequence in Table 1 describes the sequence of purchases made by a customer identified by s_1: he has purchased products a and d, then purchased product b.

In the following, we set the minimum support minSup to 0.5. *Let us consider the sequence $s = \langle(a)(b)\rangle$. Its support in $\mathcal{D}$ is $sup_{\mathcal{D}}(s) = 8$. So, $sup_{\mathcal{D}}(s) \geq minSup \times |\mathcal{D}|$, thus s is a sequential pattern in $\mathcal{D}$.*

Why Taking into Account Additional Information?

Considering the previous example, the available contextual information is the age and the gender of customers. A context could be *young female* or *old customer (for any gender)*. Therefore, when considering traditional sequential pattern mining (SPM) on data enriched with contextual information we encounter the following drawbacks.

1. **Some context-dependent behaviors are wrongly considered as general, although they are frequent in only one subcategory of customers.** For instance, $s = \langle(a)(b)\rangle$ is a sequential pattern in $\mathcal{D}$. However, by studying carefully the sequence database, we easily note that s is much more specific to *young* persons. Indeed, 7 out of 8 *young* customers support this sequence, while only 1 out of 6 *old* customers follows this pattern. This problem is directly related to how data cover the customer categories. The fact that young customers are more numerous in the database than old customers allows for a sequence being frequent in young customers only to be frequent in the whole database. However, an expert

studying context-dependent patterns in the whole database does not want a pattern being frequent in young customers only to be considered representative in the whole database.

2. **A sequential pattern extracted in a given population does not bring any information about the rest of the population.** For instance, an expert studying frequent behaviors in the *young customers* population will extract the sequence $s = \langle (a)(b) \rangle$. However, the only information provided by this sequential pattern is that *young* customers frequently follow this behavior. An expert can be interested in more information: "*Is this behavior also frequent in the rest of the population or is it exclusively specific to young people?*". Morevover, please note that mining emerging patterns in the young customers population is not a solution here. Indeed, as pointed out for the frequency constraint, if a pattern is emerging in the young customers context compared to the rest of the population, it does not guarantee that it is an emerging pattern for every type of young people.

These drawbacks show that traditional SPM is not relevant when behavior depends on contextual information associated with data sequences. We describe in the following how contextual information are formally handled through mining context-dependent sequential patterns.

3 Context-Dependent Sequential Patterns

This section proposes a formal description of contextual data, and defines the different types of context-dependent sequential patterns we aim to mine.

3.1 Contextual Sequence Database

We define a ***contextual sequence database*** $\mathcal{CD}$ as a relation $\mathcal{R}(ID, S, D_1, \ldots D_n)$, where $dom(S)$ is a set of sequences and $dom(D_i)$ for $1 \leq i \leq n$ is the set of all possible values for D_i. D_1, D_2, $\ldots D_n$ are called the ***contextual dimensions*** in $\mathcal{CD}$. A *tuple* $u \in \mathcal{CD}$ is denoted by $\prec id, s, d_1, \ldots, d_n \succ$.

Values on contextual dimensions can be organized as hierarchies. For $1 \leq i \leq n$, $dom(D_i)$ can be extended to $dom'(D_i)$, where $dom(D_i) \subseteq dom'(D_i)$. Let $\subseteq_{D_i}$ be a partial order such that $dom(D_i)$ is the set of minimal elements of $dom'(D_i)$ with respect to $\subseteq_{D_i}$. Then, the partially ordered set $(dom'(D_i), \subseteq_{D_i})$ is the ***hierarchy on dimension*** D_i, denoted by $\mathcal{H}_{D_i}$.

Example 2. *We consider $\mathcal{H}_{Age}$ and $\mathcal{H}_{Gender}$ the hierarchies on dimensions Age and Gender given in Figure 1.*

In this example, $dom(Age) = \{young, old\}$ and $dom'(Age) = dom(Age) \cup \{*\}$. The partial order $\subseteq_{Age}$ is defined such that young $\subseteq_{Age} *$ and old $\subseteq_{Age} *$.

Similarly, $dom(Gender) = \{male, female\}$ *and* $dom'(Gender) = dom(Gender) \cup \{*\}$. *The partial order* $\subseteq_{Gender}$ *is defined such that male* $\subseteq_{Gender} *$ *and* $female \subseteq_{Gender} *$.

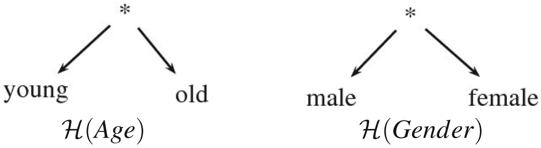


Fig. 1 Hierarchies on dimensions *Age* and *Gender*

A ***context*** c in $\mathcal{CD}$ is denoted by $[d_1, \ldots d_n]$ where $d_i \in dom'(D_i)$. If, for $1 \leq i \leq n$, $d_i \in dom(D_i)$, then c is called a ***minimal context***.

Let c_1 and c_2 be two contexts in $\mathcal{CD}$, such that $c_1 = [d_1^1, \ldots, d_n^1]$ and $c_2 = [d_1^2, \ldots d_n^2]$. Then $c_1 \leq c_2$ iff $\forall i$ with $1 \leq i \leq n$, $d_i^1 \subseteq_{D_i} d_i^2$. Moreover, if $\exists i$ with $1 \leq i \leq n$ such that $d_i^1 \subset_{D_i} d_i^2$, then $c_1 < c_2$. In this case, c_1 is said then to be ***more specific*** than c_2, and c_2 is ***more general*** than c_1.

In addition, if $c_1 \not\geq c_2$ and $c_1 \not\leq c_2$, then c_1 and c_2 are **incomparable**.

Example 3. *In Table 1, there are four minimal contexts:* $[y,m]$, $[y,f]$, $[o,m]$, *and* $[o,f]$, *where y and o respectively stand for young and old, and m and f respectively stand for male and female. In addition, context* $[*,*]$ *is more general than* $[y,*]$ *(i.e.,* $[*,*] > [y,*]$*). On the other hand,* $[y,*]$ *and* $[*,m]$ *are incomparable.*

The set of all contexts associated with the partial order $\leq$ is called the ***context hierarchy*** and denoted by $\mathcal{H}$. Given two contexts c_1 and c_2 such that $c_1 > c_2$, c_1 is called an ***ancestor*** of c_2, and c_2 is a ***descendant*** of c_1.

For instance, Figure 2 shows a representation of $\mathcal{H}$ for data provided in Table 1 and hierarchies previously given for dimensions *Age* and *Gender*.

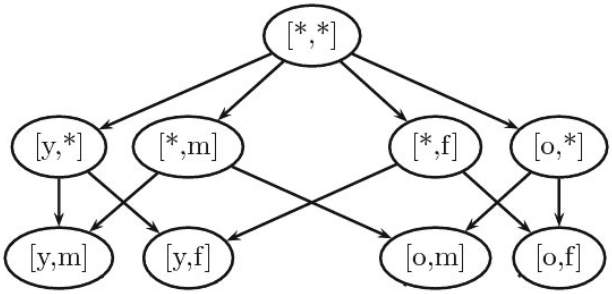


Fig. 2 The context hierarchy $\mathcal{H}$

Let us now consider the tuples $u = \prec id, s, d_1, \ldots d_n \succ$ of $\mathcal{CD}$ according to contexts defined above. The context $c = [d_1, \ldots d_n]$ is called the ***context of*** u. Note that the context of u is minimal ($\forall i$ with $1 \leq i \leq n$, $d_i \in dom(D_i)$).

Let u be a tuple in $\mathcal{CD}$ and c the context of u. For all contexts c' such that $c' \geq c$ we say that c' ***contains*** u (and u is contained by c').

Let c be a context (not necessarily minimal) in $\mathcal{CD}$. The ***sequence database of*** c, denoted by $\mathcal{D}(c)$, is the set of tuples contained by c. We define the ***size*** of a context c, denoted by $|c|$, as the size of its sequence database, i.e., $|c| = |\mathcal{D}(c)|$.

Example 4. *In Table 1, let us consider contexts $[o,m]$ and $[o,*]$. Then $\mathcal{D}([o,m]) = \{s_9, s_{10}, s_{11}\}$ and $\mathcal{D}([o,*]) = \{s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}\}$.*

Thus, $|[o,m]| = 3$ and $|[o,*]| = 6$.

3.2 Context-Dependent Sequential Patterns

The previous section showed how a contextual sequence database can be divided into several sequence databases according to contexts.

In the following, we consider the context c and the sequence s.

Definition 1 (c-frequency). *Sequence s is **frequent in** c (c-frequent) iff s is frequent in $\mathcal{D}(c)$, i.e., if $sup_{\mathcal{D}(c)}(s) \geq minSup \times |c|$. We also say that s is a **sequential pattern** in c. In the following, for simplicity, we note $sup_{\mathcal{D}(c)}(s)$ by $sup_c(s)$.*

As seen in Section 2, we focus on mining sequential patterns that, regarding contextual information, are of interest in a given context. We define two different types of patterns that we aim to mine for a given context: general patterns and exclusive patterns.

Definition 2 (c-generality). *Sequence s is **general in c** (c-general) iff:*

1. *s is frequent in c.*
2. *s is frequent in every descendant of c in the context hierarchy.*

The c-generality property ensures that a sequential pattern frequent in a given context is also frequent in all its descendants. Hence, such patterns are not sensitive to the problem of data repartition over contexts highlighted in Section 2. For instance, the sequence $\langle(a)(b)\rangle$ that is frequent in the whole database (i.e., the context $[*,*]$) is not general because it is not frequent in the old customers context. Please note that the set of general sequential patterns in a context is included in the set of sequential patterns.

Example 5. *Table 2 shows, from the contextual sequence database provided in Table 1, the sequences being frequent in at least one minimal context as well as their support for each minimal context (of the form $sup_c(s)/|\mathcal{D}(c)|$). When the support is displayed in bold, then the sequence is frequent in the corresponding minimal context.*

Table 2 Sequential patterns in minimal contexts of $\mathcal{CD}$

sequence	$[y,m]$	$[y,f]$	$[o,m]$	$[o,f]$
$\langle(a)\rangle$	**5/5**	**3/3**	**2/3**	**2/3**
$\langle(b)\rangle$	**5/5**	**3/3**	**3/3**	**3/3**
$\langle(d)\rangle$	2/5	1/3	**3/3**	**2/3**
$\langle(e)\rangle$	0/5	0/3	0/3	**3/3**
$\langle(a)(b)\rangle$	**5/5**	**2/3**	1/3	0/3
$\langle(b)(a)\rangle$	0/5	**2/3**	**2/3**	**2/3**
$\langle(bd)\rangle$	1/5	0/3	**3/3**	**2/3**

Let us now consider the context $[o,*]$ (corresponding to old people). According to Definition 2, a sequence s is general in the context $[o,*]$ iff s is frequent in $[o,*]$ (i.e., the context itself), $[o,m]$ and $[o,f]$ (i.e., its descendants in the context hierarchy).

All sequences $\langle(a)\rangle$, $\langle(b)\rangle$, $\langle(d)\rangle$, $\langle(b)(a)\rangle$ and $\langle(bd)\rangle$ satisfy these conditions. They are $[o,*]$-general. On another hand, sequence $\langle(e)\rangle$ is frequent in $[o,*]$ (it is supported by 3 old customers of 6) but not in its descendant $[o,m]$. In consequence, $\langle(e)\rangle$ is not general in $[o,*]$.

However, c-generality only considers whether a given sequential pattern in a context is frequent in its descendants, without considering the rest of the context hierarchy. We therefore propose c-exclusive sequential patterns, by considering whether there exists another context in the rest of the hierarchy (except c and its descendants) where s is general.

Definition 3 (c-exclusivity). *Sequence s is* ***exclusive in c*** *(c-exclusive) iff:*

1. *s is general in c.*
2. *there does not exist a context $c' \nleq c$ such that s is c'-general.*

In other words, a c-exclusive sequential pattern is general in c and c's descendants only. That can be seen as a discriminance constraint w.r.t. the c-generality property.

Example 6. *According to definition 3, a sequence s is exclusive in the context $[o,*]$ iff s is general in $[o,*]$, $[o,m]$ and $[o,f]$ only. Given the sequences being general in $[o,*]$, we can note that only two of them meet this requirement: $\langle(d)\rangle$ and $\langle(bd)\rangle$. On another hand, sequence $\langle(b)(a)\rangle$ is also general in $[*,f]$ and therefore is not exclusive in $[o,*]$.*

We have defined general and exclusive sequential patterns, two types of context-dependent sequential patterns. Those patterns are frequent sequences in a context c that satisfy some interesting properties regarding contextual information. Such sequences can be exploited to assist a user in a context-driven analysis of data.

4 Mining Context-Dependent Sequential Patterns

In this section, we detail the properties that will help us to efficiently mine context-dependent patterns defined in previous section.

4.1 Preliminary Definitions and Properties

In the following, we will mainly rely on the minimal contexts of the hierarchy. In order to easily manipulate these elements, we define the decomposition of a context as the set of its minimal descendants.

Definition 4 (decomposition of a context). *Let c be a context. The decomposition of c in $\mathcal{CD}$, denoted by $decomp(c)$, is the non-empty set of minimal contexts such that $\forall c' \in decomp(c)$, $c \geq c'$.*

Example 7. *The decomposition of context $[y,*]$ is $\{[y,m],[y,f]\}$.*

The decomposition of a context c forms a partition of its sequence database. Consequently, we can highlight some immediate set-theoretical properties.

Lemma 1. *Let c be a context, $decomp(c) = \{c_1, c_2, \ldots, c_n\}$ its decomposition and s a sequence. Given the definition of the sequence database of c, the decomposition of c has the following properties:*

1. $\bigcap\limits_{i=1}^{n} \mathcal{D}(c_i) = \emptyset$;
2. $\bigcup\limits_{i=1}^{n} \mathcal{D}(c_i) = \mathcal{D}(c)$;
3. $|c| = |\mathcal{D}(c)| = \sum\limits_{i=1}^{n} |c_i|$;
4. $sup_c(s) = \sum\limits_{i=1}^{n} sup_{c_i}(s)$.

Lemma 1 can be exploited to unveil an interesting property about the c-frequency of sequential patterns in the decomposition of c.

Lemma 2. *Let c be a context, and $decomp(c) = \{c_1, c_2, \ldots, c_n\}$. If $\forall i \in \{1,\ldots,n\}$, s is frequent in c_i, then s is frequent in c. In addition, s is frequent in all the descendants of c.*

Proof: *For each c_i such that $i \in \{1,..n\}$, $sup_{c_i}(s) \geq minSup \times |c_i|$. This means $\sum\limits_{i=1}^{k} sup_{c_i}(s) \geq \sum\limits_{i=1}^{n} minSup \times |c_i|$. However, $\sum\limits_{i=1}^{n} minSup \times |c_i| = minSup \times \sum\limits_{i=1}^{n} |c_i| = minSup \times |c|$. Since $\sum\limits_{i=1}^{k} sup_{c_i}(s) = sup_c(s)$ it follows $sup_c(s) \geq minSup \times |c|$.*

Let c' be a context such that $c > c'$. Then $decomp(c') \subseteq decomp(c)$, i.e., s is frequent in all contexts in $decomp(c')$. Applying the previous result we obtain s, a frequent sequence in c'. □

In the following, we show how we benefit from such properties in order to efficiently mine context-dependent patterns.

We first have interest in mining c-general sequential patterns. Given the definition of a c-general sequential pattern, a naive approach could be performed in the following steps:

1. Mine sequential patterns in c.
2. Mine sequential patterns in every descendants of c.
3. Output sequential patterns frequent in c and all its descendants.

However, the number of contexts to mine can be very large (i.e., c and c's descendants). In order to overcome this drawback, we exploit the properties of the context hierarchy and show that c-general sequential patterns can be extracted by focusing only on the decomposition of c.

Theorem 1. *The sequence s is c-general iff $\forall c' \in decomp(c)$, s is frequent in c'.*

Proof: *If s is frequent in each context of $decomp(c)$, then, by applying Lemma 2, s is frequent in c and c's descendants in the context hierarchy, i.e., it is general in c. In addition, if $\exists c' \in decomp(c)$ such that s is not frequent in c', then there exists a descendant of c where s is not frequent and s is not general in c according to Definition 2.* □

Theorem 1 is a key result as it guarantees that c-general sequential patterns are sequences being frequent in all minimal descendants of c. This property can therefore be exploited in order to mine general sequential patterns in a context c. Moreover, c-generality provides us with the following lemma.

Lemma 3. *If a sequence s is not c-general, then $\forall s'$ such that $s' \sqsupseteq s$, s' is not c-general.*

Proof: *If s is not c-general, then there exists a context $c' \leq c$ where s is not frequent. Given s' a sequence such that $s' \sqsupseteq s$, s' is also not frequent in c'. As a result, s' is not c-general.* □

Lemma 3 shows that c-generality is anti-monotonic with respect to the size of the sequence. This property will be useful when coming to extract c-general sequential patterns.

We also aim at mining c-exclusive sequential patterns. A naive approach to mine such patterns could be done in the following steps:

1. Mine general sequential patterns in c (see previous naive approach).
2. Mine general sequential patterns in each context c' of the hierarchy that is not c or a descendant of c.

3. Output general sequential patterns in c that are not general in any other context c'.

This approach is very time consuming, as it requires to mine sequential patterns in all contexts of the hierarchy. However, a similar reasoning as for c-general sequential patterns can be applied to redefine the c-exclusivity property.

Lemma 4. *There exists a context $c' \nleq c$ such that s is c'-general iff there exists a minimal context c'' such that $c'' \notin decomp(c)$ and s is frequent in c''.*

Proof: *If s is c'-general, then s is frequent in each element of $decomp(c')$. However, if $c' \nleq c$, then at least one element of $decomp(c')$ is not an element of $decomp(c)$. So, there exists a minimal context c'' such that $c'' \notin decomp(c)$ and s is frequent in c''.*

Moreover, if there exists a minimal context c'' such that $c'' \notin decomp(c)$ and s is frequent in c'', then s is c''-general. However, $c'' \nleq c$. As a result, there exists a context $c' \nleq c$ such that s is c'-general. □

Theorem 2. *Let $\mathcal{M}$ be the set of minimal contexts in $\mathcal{H}$. The sequence s is c-exclusive iff:*

1. $\forall c' \in decomp(c)$, s is frequent in c',
2. $\forall c'' \in \mathcal{M} \setminus decomp(c)$, s is not frequent in c''.

Proof: *This result is obtained by directly applying Theorem 1 and Lemma 4 to the definition of a c-exclusive sequential pattern (Definition 3).* □

Hence, a c-exclusive sequential pattern is a c-general sequential pattern s such that there does not exist a minimal context outside the decomposition of c where s is frequent.

Hence, Theorems 1 and 2 show that both general and exclusive sequential patterns can be mined by considering minimal contexts only, while naive approaches require to consider all descendants of c to extract c-general sequential patterns, and all the contexts of the hierarchy to mine c-exclusive sequential patterns. In the following, we propose an algorithm that exploits these results in order to efficiently mine context-dependent sequential patterns.

4.2 Algorithm

This section presents *Gespan*, an algorithm designed to mine both general and exclusive sequential patterns in a given context. It is based on the *PrefixSpan* algorithm [Pei et al., 2004] that aims at solving traditional sequential pattern mining. We explain the principles of *PrefixSpan* in the following example, by describing the process of mining sequential patterns in the sequence database $\mathcal{D}$ from Table 1, with a minimum support threshold set to *0.5*.

Example 8. *A scan of the sequence database extracts all the sequential patterns of the form $\langle(i)\rangle$, where i is an item. Hence, PrefixSpan finds $\langle(a)\rangle$, $\langle(b)\rangle$, $\langle(d)\rangle$, since $\langle(c)\rangle$ and $\langle(e)\rangle$ are not frequent.*

In consequence, the whole set of sequential patterns in $\mathcal{D}$ can be partitioned into subsets, each subset being the set of sequential patterns having $\langle(i)\rangle$ as a prefix. These subsets can be extracted by mining the ***projected databases*** *for each prefix, i.e., for each $\langle(i)\rangle$. A projected database contains, for each data sequence, its subsequence containing all frequent items following the first occurrence of the given prefix. Such a subsequence is called a* ***postfix***. *If the first item x of the postfix is in the same itemset as the last item of the prefix, the postfix is denoted by $\langle(\_x\dots)\dots\rangle$.*

Then, $\langle(a)\rangle$ is outputted, and the $\langle(a)\rangle$-projected database is built, containing 11 postfixes: $\langle(\_d)(b)\rangle$, $\langle(\_b)(b)\rangle$, $\langle(a)(b)\rangle$, $\langle(bc)\rangle$, etc. Then items i, such that either $\langle(ai)\rangle$ or $\langle(a)(i)\rangle$ is frequent, are extracted from the $\langle(a)\rangle$-projected database. b is such an item, as $\langle(a)(b)\rangle$ is a sequential pattern. So, the process continues by outputting $\langle(a)(b)\rangle$, and using it as a new prefix.

We now present the *Gespan* algorithm that aims at mining general and exclusive sequential patterns in a context.

The prefix-growth approach of PrefixSpan is used to extract general sequential patterns, relying on the anti-monotonicity of the c-generality property. From a prefix sequence s, the algorithm builds the s-projected database by making use of the method $BuildProjectedDatabase$, and scans the projected database (method *ScanDB*) to find items i that can be assembled to form a new general sequential pattern s'. Then, the s'-projected database is built and the process continues. Since the general intuition is similar to *PrefixSpan*, we do not detail the *ScanDB* and *BuildProjectedDatabase* methods of *Gespan*, but only focus on the differences.

In method *ScanDB*$(\mathcal{CD})$, the support of i is computed in each minimal context of the s-projected database. Testing the c-generality of the resulting sequence s' in the projected database is based on Theorem 1 and performed by method *isGeneral*$(s',C,\mathcal{H})$ described in Algorithm 2.

Then, for each c-general sequential pattern, method *isExclusive*$(s',C,\mathcal{H})$ described in Algorithm 3 is used to test whether this sequential pattern is also c-exclusive. Please note that if the user is only interested in mining c-general sequential patterns, this step of the algorithm can be removed.

5 Experiments

All experiments have been performed on a system equipped with a 3GHz CPU and 16GB of main memory. The methods are implemented in C++.

By conducting these experiments, we wish to evaluate the performances of *Gespan* by focusing on two aspects.

Number of Patterns. We study the number of context-dependent sequential patterns extracted with *Gespan*, and compare it to the number of frequent sequences

Algorithm 1. Gespan

```
Input: $\mathcal{CD}$ a contextual sequence database, $minSup$ a minimum support threshold, $\mathcal{H}$ a context hierarchy, $C$ a context in $\mathcal{H}$.
  Call $subGespan(\langle\rangle, \mathcal{CD}, \mathcal{H}, C)$;
```

```
Subroutine subGespan($s$,$\mathcal{CD}$,$\mathcal{H}$,$C$)
Input: $s = \langle I_1 \dots I_n \rangle$ a sequence; $\mathcal{CD}$ the $s$-projected database, $\mathcal{H}$ a context hierarchy, $C$ a context in $\mathcal{H}$.
  $ScanDB(\mathcal{CD})$
  Let $\mathcal{I}$ be the set of items $i$ such that $isGeneral(\langle I_1 \dots (I_n \cup i)\rangle, C, \mathcal{H})$ returns $TRUE$
  Let $\mathcal{I}'$ be the set of items $i$ such that $isGeneral(\langle I_1 \dots I_n(i)\rangle, C, \mathcal{H})$ returns $TRUE$
  for all $i \in (\mathcal{I} \cup \mathcal{I}')$ do
    $s'$ is the sequence such that $i$ is appended to $s$
    if isExclusive($s'$, $C$, $\mathcal{H}$) then
      output $s'$ as a $C$-exclusive sequential pattern
    end if
    output $s'$ as a $C$-general sequential pattern
    $\mathcal{CD}' = BuildProjectedDatabase(s', \mathcal{CD})$
    call $subGespan(s', \mathcal{CD}', \mathcal{H})$
  end for
```

Algorithm 2. isGeneral(s, C, $\mathcal{H}$)

```
Input: $s$ a sequence, $C$ a context, $\mathcal{H}$ a context hierarchy.
  for all $c \in decomp(C)$ do
    if $s$ is not frequent in $c$ then
      return FALSE
    end if
  end for
  return TRUE
```

Algorithm 3. isExclusive(s, C, $\mathcal{H}$)

```
Input: $s$ a pattern, $C$ a context, $\mathcal{H}$ a context hierarchy.
  Let $\mathcal{M}$ be the set of minimal contexts in $\mathcal{H}$
  for all $c \in \mathcal{M} \setminus decomp(C)$ do
    if $s$ is frequent in $c$ then
      return FALSE
    end if
  end for
  return TRUE
```

(i.e., sequential patterns) extracted with *PrefixSpan*. Indeed, we show in Section 2 that some traditional sequential patterns are irrelevant when considering the analysis of contextual data. This experiment will allow to quantify this aspect.

Runtime. We measure the execution time required to mine context-dependent patterns in a given context, and compare it to the time required to mine frequent sequences in the same context.

5.1 Data Description

The experiments were conducted on about 100000 product reviews from *amazon.com*, in order to study the vocabulary used according to reviews. This dataset is a subset of the one used in [Jindal and Liu, 2008]. Reviews have been lemmatized[1] and grammatically filtered in order to remove uninteresting terms, by using the *tree tagger* tool [Schmid, 1994]. Preserved terms are verbs (apart from modal verbs and the verb *"to be"*), nouns, adjectives and adverbs. Remaining terms have been stemmed[2] using the Porter algorithm [Porter, 1980]. Then, the sequence database is constructed using the following principles:

- each review is a sequence,
- each sentence is an itemset (i.e., the order of the words in a sentence is not considered),
- each word is an item.

An extracted sequential pattern could be $\langle(eat\ mushroom)(hospital)\rangle$, which means that frequently a review contains *eat* and *mushroom* in a sentence and *hospital* in one of the following sentences.

Contextual Dimensions

Each review is associated with contextual dimensions:

- the *product* type (*Books*, *DVD*, *Music* or *Video*)
- the *rating* (originally a numeric value r between 0 and 5). For these experiments, r has been translated into qualitative values: *bad* (if $0 \leq r < 2$), *neutral* (if $2 \leq r \leq 3$), and *good* (if $3 < r \leq 5$)
- the proportion of helpful *feedbacks*[3], i.e., *0-25%*, *25-50%*, *50-75%* or *75-100%*.

We define hierarchies on contextual dimensions as described in Figure 3. The number of contexts in the context hierarchy is $|dom'(product)| \times |dom'(rating)| \times |dom'(feedbacks)| = 6 \times 5 \times 7 = 210$, while the number of minimal contexts is $|dom(product)| \times |dom(rating)| \times |dom(feedbacks)| = 4 \times 3 \times 4 = 48$.

Note that the domain of values of contextual dimensions has been enriched with new values. For instance, hierarchy $\mathcal{H}(rating)$ contains an element $Extreme$ that

[1] i.e., the different forms of a word have been grouped together as a single item. For instance, the different forms of the verb *to be* (is, are, was, being, etc.) are all returned as *to be*.

[2] i.e., the inflected forms of a word are reduced to their root form. For instance, the adjective *musical* is returned as *music*.

[3] On amazon.com each reader can post a feedback on a review.

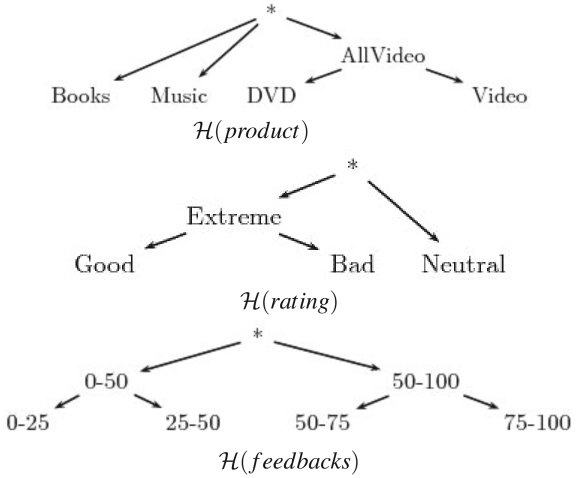


Fig. 3 Hierarchies on contextual dimensions

will allow us, for instance, to extract patterns being general in extreme opinions (positive or negative).

5.2 Results and Discussion

It is not possible to show the obtained results for each of the 210 contexts in the hierarchy. As a consequence, we will provide the results for a selection of more or less general contexts:

- $[*,*,*]$ is the more general context of the hierarchy. It corresponds to all the reviews in the database.
- $[Books,*,*]$ is the context corresponding to all the reviews that are related to a book.
- $[Books,bad,*]$ is a more specific context than $[Books,*,*]$. It corresponds to bad reviews associated to a book.
- $[Books,bad,75-100]$ is a minimal context in the hierarchy. It corresponds to bad reviews associated to a book that have been considered useful by more than 75% of voting amazon users.

Table 3 presents the number of patterns extracted with $minSup = 0.01$ for each algorithm: *PrefixSpan* to extract sequential patterns (i.e., frequent sequences), *Gespan* to extract general sequential patterns (GSP) and exclusive sequential patterns (ESP). First, please note that the number of general sequential patterns is significantly lower than the number of sequential patterns in all non-minimal contexts. This shows that a large proportion of patterns that are frequent in a context are actually specific to a sub-part of this context only. However, the number of general sequential patterns in

Table 3 Number of sequential patterns, general sequential patterns and exclusive sequential patterns, according to the context, for $minSup = 0.01$

Context	PrefixSpan	Gespan (GSP)	Gespan (ESP)
$[*, *, *]$	50089	1788	1788
$[Books, *, *]$	79113	15147	5179
$[Books, bad, *]$	194765	62661	603
$[Books, bad, 75-100]$	259790	259790	19801

$[Books, bad, 75-100]$ is equal to the number of sequential patterns. Indeed, because minimal contexts have no descendants, a sequential pattern in a minimal context is general in this context.

Second, the number of exclusive sequential patterns is significantly lower than the number of general sequential patterns in all contexts, except for $[*, *, *]$. For instance, only 1% of general sequential patterns in $[Books, bad, *]$ is actually exclusive in this context. However, the number of exclusive sequential patterns in $[*, *, *]$ is equal to the number of general sequential patterns. Indeed, there does not exist a context that is not a descendant of $[*, *, *]$. As a result, all general sequential patterns in $[*, *, *]$ are also exclusive.

These results are directly related to the definition of frequent, general and exclusive sequential patterns. General sequential patterns are indeed frequent sequential patterns that satisfy the representativity constraint required by the *c*-generality. As a consequence, the set of general patterns in a context is included in the set of frequent patterns. Similarly, exclusive sequential patterns are general sequential patterns in a context satisfying an additional constraint (being general in this context and its sub-contexts only). The set of exclusive patterns in a context is therefore included in the set of general patterns of the same context.

Table 4 Runtime in seconds for extracting each type of sequential patterns, according to the context, for $minSup = 0.01$

Context	PrefixSpan	Gespan (GSP)	Gespan (GSP + ESP)
$[*, *, *]$	1795	131	131
$[Books, *, *]$	1435	463	646
$[Books, bad, *]$	449	216	1344
$[Books, bad, 75-100]$	212	212	2477

Table 4 shows the execution time needed to mine each type of sequential patterns. Two versions of *Gespan* have been used. The first one aims at mining general sequential patterns only, while the second aims at mining both general and exclusive sequential patterns. Mining general sequential patterns in non-minimal contexts is always faster than mining sequential patterns. We also note that the gap in the runtime is larger when the considered context is more general.

The time required to extract exclusive sequential patterns strongly depends on the level of generalization of the mined context. Indeed, when the considered context is very general (e.g., $[*,*,*]$ or $[Books,*,*]$) then mining exclusive sequential patterns is faster than mining sequential patterns. However, for more specific contexts (e.g., $[Books, bad, *]$ or $[Books, bad, 75-100]$) the mining of exclusive sequential patterns is time consuming. This is due to the number of minimal contexts that must be considered in order to test the exclusivity of a sequential pattern. For instance, in order to test whether a general sequential pattern s is exclusive in $[Books, bad, 75-100]$, *Gespan* needs to check whether s is frequent in one of the 47 other minimal contexts in the hierarchy. This number of minimal contexts is lower when the considered context is more general. Hence, mining exclusive sequential pattern is more efficient in more general contexts.

In addition, please note that we have not compared *Gespan* with the baseline approaches described in Section 4, but only with *PrefixSpan*. The reason is that baseline approaches are very naive, and obviously more time-consuming than *PrefixSpan*. Moreover, the comparison with *PrefixSpan* allows us to confront the two advantages of *Gespan* over a traditional sequential pattern mining algorithm. First, general or exclusive sequential patterns are more informative than frequent sequential patterns, as they consider only representative patterns when contextual information is available. Second, *Gespan* exploits theoretical properties highlighted in Section 4 and offers reduced runtimes (except for mining exclusive patterns in very specific contexts, as shown in Table 4).

6 Conclusion

In this paper we have motivated the need for mining context-dependent sequential patterns in a sequence database enriched with contextual information. We formally defined the problem and unveiled set-theoretical properties that allow database mining in a concise manner.

This work can be extended in a number of ways. First, in this paper we have specifically handled sequential patterns. An immediate prospect is the generalization of this work to other types of frequent patterns such as frequent episodes [Mannila et al., 1997] or frequent subgraphs [Kuramochi and Karypis, 2001]. Second, we have only focused on a minimum support threshold to extract context-dependent sequential patterns. In future work, we aim at studying other constraints. For instance, we have already pointed out in Section 1 that mining general and exclusive patterns can be seen as related to the problem of mining emerging patterns. An interesting prospect consists in mining context-dependent emerging patterns by adapting the corresponding constraint to the notion of c-generality and c-exclusivity defined for context-dependent sequential patterns.

References

[Agrawal et al., 1993] Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. SIGMOD Rec. 22(2) (1993)

[Agrawal and Srikant, 1995] Agrawal, R., Srikant, R.: Mining sequential patterns. In: Yu, P.S., Chen, A.S.P. (eds.) Eleventh International Conference on Data Engineering. IEEE Computer Society Press (1995)

[Dong and Li, 1999] Dong, G., Li, J.: Efficient mining of emerging patterns: discovering trends and differences. In: KDD 1999: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York (1999)

[Dong et al., 1999] Dong, G., Zhang, X., Wong, L., Li, J.: CAEP: Classification by Aggregating Emerging Patterns. In: Arikawa, S., Nakata, I. (eds.) DS 1999. LNCS (LNAI), vol. 1721, pp. 30–42. Springer, Heidelberg (1999)

[Hilderman et al., 1998] Hilderman, R.J., Carter, C.L., Hamilton, H.J., Cercone, N.: Mining Market Basket Data Using Share Measures and Characterized Itemsets. In: Wu, X., Kotagiri, R., Korb, K.B. (eds.) PAKDD 1998. LNCS, vol. 1394, Springer, Heidelberg (1998)

[Jindal and Liu, 2008] Jindal, N., Liu, B.: Opinion spam and analysis. In: Proceedings of the International Conference on Web Search and Web Data Mining. ACM (2008)

[Kuramochi and Karypis, 2001] Kuramochi, M., Karypis, G.: Frequent subgraph discovery. In: Proceedings IEEE International Conference on Data Mining, ICDM 2001, pp. 313–320. IEEE (2001)

[Li et al., 2001] Li, J., Dong, G., Ramamohanarao, K.: Making use of the most expressive jumping emerging patterns for classification. Knowledge and Information Systems 3(2), 131–145 (2001)

[Mannila et al., 1997] Mannila, H., Toivonen, H., Inkeri Verkamo, A.: Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery 1(3), 259–289 (1997)

[Pei et al., 2004] Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: Mining sequential patterns by pattern-growth: the PrefixSpan approach. IEEE Transactions on Knowledge and Data Engineering 16(11) (2004)

[Pinto et al., 2001] Pinto, H., Han, J., Pei, J., Wang, K., Chen, Q., Dayal, U.: Multidimensional sequential pattern mining. In: Proceedings of the Tenth International Conference on Information and Knowledge Management. ACM (2001)

[Plantevit et al., 2005] Plantevit, M., Choong, Y.W., Laurent, A., Laurent, D., Teisseire, M.: M^2SP: Mining Sequential Patterns Among Several Dimensions. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS (LNAI), vol. 3721, pp. 205–216. Springer, Heidelberg (2005)

[Porter, 1980] Porter, M.: An algorithm for suffix stripping. Program: Electronic Library & Information Systems 40(3), 211–218 (1980)

[Rabatel et al., 2010] Rabatel, J., Bringay, S., Poncelet, P.: Contextual Sequential Pattern Mining. In: 2010 IEEE International Conference on Data Mining Workshops, pp. 981–988. IEEE (2010)

[Schmid, 1994] Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: Proceedings of International Conference on New Methods in Language Processing, vol. 12. Citeseer (1994)

[Ziembiński, 2007] Ziembiński, R.: Algorithms for context based sequential pattern mining. Fundamenta Informaticae 76(4), 495–510 (2007)

Comparison of Proximity Measures: A Topological Approach

Djamel Abdelkader Zighed, Rafik Abdesselam, and Ahmed Bounekkar

Abstract. In many application domains, the choice of a proximity measure affect directly the result of classification, comparison or the structuring of a set of objects. For any given problem, the user is obliged to choose one proximity measure between many existing ones. However, this choice depend on many characteristics. Indeed, according to the notion of equivalence, like the one based on pre-ordering, some of the proximity measures are more or less equivalent. In this paper, we propose a new approach to compare the proximity measures. This approach is based on the topological equivalence which exploits the concept of local neighbors and defines an equivalence between two proximity measures by having the same neighborhood structure on the objects. We compare the two approaches, the pre-ordering and our approach, to thirty five proximity measures using the continuous and binary attributes of empirical data sets.

1 Introduction

Comparing objects, situations or things leads to identifying and assessing hypothesis or structures that are related to real objects or abstract matters. In other words, for understanding situations that are represented by a set of objects and be able to act

Djamel Abdelkader Zighed · Rafik Abdesselam
Laboratoire ERIC, University Lumière of Lyon 2,
5 Avenue Pierre Mendès-France, 69676 Bron Cedex, France
e-mail: {abdelkader.zighed,rafik.abdesselam}@univ-lyon2.fr
http://eric.univ-lyon2.fr/~zighed,
http://eric.univ-lyon2.fr/~rabdesselam/fr/

Ahmed Bounekkar
Laboratoire ERIC, University Claude Bernard of Lyon 1,
43 boulevard 11 novembre 1918, 69622 Villeurbanne Cedex, France
e-mail: ahmed.bounekkar@univ-lyon1.fr
http://eric.univ-lyon2.fr/

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 43–58.
DOI: 10.1007/978-3-642-35855-5_3

upon, we must be able to compare them. In natural life, this comparison is achieved unconsciously by the brain. In the artificial intelligence context we should describe how the machine might perform this comparison. One of the basic element that we have to specify, is the proximity measure between objects.

The proximity measures are characterized by a set of mathematical properties. The main objects, that we seek to explain in this paper, are how we can assess and which measure we can use to prove: are two specifics proximity measures equivalent or not? What is the meaning of equivalence between two proximity measures? In which situation can we consider that two proximity measures are equivalent? If two measures are equivalent, does it means that they are substitutable between each other? Does the choice of a specific proximity measure between individuals immersed in a multidimensional space, like $\mathbb{R}^p$, influence or not the result of clustering or k-nearest neighbors? These objects are important in many practical applications such as retrieval information area. For instance, when we submit a query to a search engine, it displays, so fast, a list of candidate's answers ranked according to the degree of resemblance to the query. Then, this degree of resemblance can be seen as a measure of dissimilarity or similarity between the query and the available objects in the database. Does the way that we measure the similarity or the dissimilarity between objects affect the result of a query? It is the same in many other areas when we seek to achieve a grouping of individuals into classes. It is obvious that the outcome of any algorithm, based on proximity measures, depends on the measure used.

A proximity measure can be defined in different ways, under assumptions and axioms that are sought, this will lead to measures with diverse and varied properties. The notion of proximity covers several meanings such as similarity, resemblance, dissimilarity, etc. In the literature, we can find a lot of measures that differ from each other depending on many factors such as the type of the used data (binary, quantitative, qualitative fuzzy...). Therefore, the choice of proximity measure remains an important issue.

Certainly, the application context, the prior knowledge, the type of data and many other factors may help in the identification of the appropriate measure. For instance, if the objects to be compared are described by Boolean vectors, we can restrict to a class of measures specifically devoted. However, the number of measure's candidates might remain quite large. In that case, how shall we proceed for identifying the one we should use? If all measure's candidates were equivalent, is it sufficient enough to take one randomly? In most cases, this is not true. The present work aims to solve this problem by comparing proximity measures. To do this, three approaches are used.

1. For example, [Richter, 1992] used, several proximity measures on the same data set and then, aggregated arithmetically their partial results into a single value. The final result can be seen as a synthesis of different views expressed by each proximity measure. This approach avoids treating the subject of the comparison which remains a problem in itself.
2. By empirical assessment: many papers describe methodologies for comparing performance of different proximity measures. To do that, we can use either

benchmarks, like in [Liu et al., , Strehl et al., 2000] where outcomes are previously known, or criteria considered as relevant and allowed the user to identifying the appropriate proximity measure. We can cite some work in this category as shown in [Noreault et al., 1980, Malerba et al., 2002, Spertus et al., 2005].

3. The objective of this paper belongs to the category of comparison proximity measures. For example, we checked if they have common properties [Lerman, 1967, Clarke et al., 2006] or if one can express as function of the other as in these references [Zhang and Srihari, 2003, Batagelj and Bren, 1995] or simply if they provide the same result by clustering operation [Fagin et al., 2003], etc. In the last case, the proximity measures can be categorized according to their degree of resemblance. The user can identify measures that are equivalent to those that are less [Lesot et al., 2009, Bouchon-Meunier et al., 1996].

We propose in this paper a new method to compare the proximity measures, which is related to the third category in order to detect those identical from the others and, to group them into classes according to their similarities. The procedure of comparing two proximity measures consists to compare the values of the induced proximity matrices [Batagelj and Bren, 1995, Bouchon-Meunier et al., 1996] and, if necessary, to establish a functional and explicit link when the measures are equivalent. For instance, to compare two proximity measures, [Lerman, 1967] focuses on the preorders induced by the two proximity measures and assess their degree of similarity by the concordance between the induced preorders by the set of pairs of objects. Other authors, such as [Schneider and Borlund, 2007b], evaluate the equivalence between two measures by a statistical test between the proximity matrices.

The numerical indicators derived from these cross-comparisons are then used to categorize measures. The common idea of these works is based on a principal that says that, two proximity measures are closer if the pre-ordering induced on pairs of objects does not change. We will give clearer definitions later.

In this paper, we propose another approach of comparing proximity measures. We introduce this approach by using the neighbors structure of objects which constitutes the main idea of our work. We call this neighborhood structure the topology induced by the proximity measure. If the neighborhood structure between objects, induced by a proximity measure u_i, does not change relatively from another proximity measure u_j, this means that the local similarities between objects do not change. In this case, we may say that the proximity measures u_i and u_j are in topological equivalence. We can thus calculate a value of topological equivalence between pairs of proximity measures and then, we can visualize the closeness between measures. This latest could be achieved by an algorithm of clustering.

We will define this new approach and show the principal links identified between our approach and the one based on preordonnance. So far, we didn't find any publication that deals with the problem in the same way as we do. The present paper is organized as follows. In section 2, we will describe more precisely the theoretical framework; in section 3, we recall the basic definitions for the approach based on the induced preordonnance; In section 4, we will introduce our approach of topological equivalence; in section 5, we will provide some evaluations of the comparison between the two approaches and will try to highlight possible links between them. The further work and open trails, provided by our approach, will be detailed in section 6, the conclusion. We will highlight some remarks, on how this work could be extended to all kind of proximity measures whatever the representation space: binary [Batagelj and Bren, 1995, Lerman, 1967, Warrens, 2008, Lesot et al., 2009], fuzzy [Zwick et al., 1987, Bouchon-Meunier et al., 1996], symbolic, [Malerba et al., 2002], etc.

2 Proximity Measures

A measure of proximity between objects can be defined as part of a mathematical properties and as the description space of objects to compare. We give, in Table 1, some conventional proximity measures defined on $\mathbb{R}^p$.

Table 1 Some measures of proximity

Measure	Formula
u_1: Euclidean	$u_E(x,y) = \sqrt{\sum_{i=1}^{p}(x_i - y_i)^2}$
u_2: Mahalanobis	$u_{Mah}(x,y) = \sqrt{(x-y)^t \sum^{-1}(x-y)}$
u_3: Manhattan (City-block)	$u_{Man}(x,y) = \sum_{i=1}^{p} \lvert x_i - y_i \rvert$
u_4: Minkowski	$u_{Min_\gamma}(x,y) = (\sum_{i=1}^{p} \lvert x_i - y_i \rvert^{\gamma})^{\frac{1}{\gamma}}$
u_5: Tchebytchev	$u_{Tch}(x,y) = \max_{1 \leq i \leq p} \lvert x_i - y_i \rvert$
u_6: Cosine Dissimilarity	$u_{Cos}(x,y) = 1 - \frac{<x,y>}{\lVert x \rVert \lVert y \rVert}$
u_7: Canberra	$u_{Can}(x,y) = \sum_{i=1}^{p} \frac{\lvert x_i - y_i \rvert}{\lvert x_i \rvert + \lvert y_i \rvert}$
u_8: Squared Chord	$u_{SC}(x,y) = \sum_{i=1}^{p}(\sqrt{x_i} - \sqrt{y_i})^2$
u_9: Weighted Euclidean	$u_{E_w}(x,y) = \sqrt{\sum_{i=1}^{p} \alpha_i (x_i - y_i)^2}$
u_{10}: Chi-square	$u_{\chi^2}(x,y) = \sum_{i=1}^{p} \frac{(x_i - m_i)^2}{m_i}$
u_{11}: Jeffrey Divergence	$u_{JD}(x,y) = \sum_{i=1}^{p}(x_i \log \frac{x_i}{m_i} + y_i \log \frac{y_i}{m_i})$
u_{12}: Histogram Intersection Measure	$u_{HIM}(x,y) = 1 - \frac{\sum_{i=1}^{p}(\min(x_i,y_i))}{\sum_{j=1}^{p} y_j}$
u_{13}: Pearson's Correlation Coefficient	$u_\rho(x,y) = 1 - \lvert \rho(x,y) \rvert$

Where, p is the dimension of space, $x = (x_i)_{i=1,\ldots,p}$ and $y = (y_i)_{i=1,\ldots,p}$ two points in $\mathbb{R}^p$, $(\alpha_i)_{i=1,\ldots,p} \geq 0$, $\sum^{-1}$ the inverse of the variance and covariance matrix, $\gamma > 0$, $m_i = \frac{x_i + y_i}{2}$ and $\rho(x,y)$ denotes the linear correlation coefficient of Bravais-Pearson.

Consider a sample of n individuals $x, y, \ldots$ in a space of p dimensions. Individuals are described by continuous variables: $x = (x_1, \ldots, x_p)$. A proximity measure u between two individuals points x and y of $\mathbb{R}^p$ is defined as follows:

$$\begin{array}{l} u : R^p \times R^p \longmapsto R \\ (x,y) \ \longmapsto \ u(x,y) \end{array}$$

with the following properties, $\forall (x,y) \in R^p \times R^p$:

$$\text{P1: } u(x,y) = u(y,x) \qquad \text{P2: } u(x,x) \geq (\leq)\, u(x,y) \qquad \text{P3: } \exists \alpha \in R \;\; u(x,x) = \alpha.$$

We can also define δ: $\delta(x,y) = u(x,y) - \alpha$ a proximity measure that satisfies the following properties, $\forall (x,y) \in R^p \times R^p$:

$$\text{T1: } \delta(x,y) \geq 0 \qquad \text{T2: } \delta(x,x) = 0 \qquad \text{T3: } \delta(x,x) \leq \delta(x,y).$$

A proximity measure that verifies properties T1, T2 and T3 is a dissimilarity measure. We can also cite other properties such as:

T4: $\delta(x,y) = 0 \Rightarrow \forall z \in R^p \;\; \delta(x,z) = \delta(y,z)$ T5: $\delta(x,y) = 0 \Rightarrow x = y$
T6: $\delta(x,y) \leq \delta(x,z) + \delta(z,y)$ T7: $\delta(x,y) \leq \max(\delta(x,z), \delta(z,y))$
T8: $\delta(x,y) + \delta(z,t) \leq \max(\delta(x,z) + \delta(y,t), \delta(x,t) + \delta(y,z))$.

Table 2 Some proximity measures for binary data

Measures: Type 1	Similarities	Dissimilarities
Jaccard (1900)	$s_1 = \frac{a}{a+b+c}$	$u_1 = 1 - s_1$
Dice (1945), Czekanowski (1913)	$s_2 = \frac{2a}{2a+b+c}$	$u_2 = 1 - s_2$
Kulczynski (1928)	$s_3 = \frac{1}{2}(\frac{a}{a+b} + \frac{a}{a+c})$	$u_3 = 1 - s_3$
Driver and Kroeber, Ochiai (1957)	$s_4 = \frac{a}{\sqrt{(a+b)(a+c)}}$	$u_4 = 1 - s_4$
Sokal and Sneath	$s_5 = \frac{a}{a+2(b+c)}$	$u_5 = 1 - s_5$
Braun-Blanquet (1932)	$s_6 = \frac{a}{\max(a+b,a+c)}$	$u_6 = 1 - s_6$
Simpson (1943)	$s_7 = \frac{a}{\min(a+b,a+c)}$	$u_7 = 1 - s_7$
Measures: Type 2		
Kendall, Sokal-Michener (1958)	$s_8 = \frac{a+d}{a+b+c+d}$	$u_8 = 1 - s_8$
Russel and Rao (1940)	$s_9 = \frac{a}{a+b+c+d}$	$u_9 = 1 - s_9$
Rogers and Tanimoto (1960)	$s_{10} = \frac{a+d}{a+2b+2c+d}$	$u_{10} = 1 - s_{10}$
Pearson ϕ (1896)	$s_{11} = \frac{ad-bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	$u_{11} = \frac{1-s_{11}}{2}$
Hamann (1961)	$s_{12} = \frac{a+d-b-c}{a+b+c+d}$	$u_{12} = \frac{1-s_{12}}{2}$
bc		$u_{13} = \frac{4bc}{(a+b+c+d)^2}$
Sokal and Sneath (1963), un_5	$s_{14} = \frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	$u_{14} = 1 - s_{14}$
Michael (1920)	$s_{15} = \frac{4(ad-bc)}{(a+d)^2+(b+c)^2}$	$u_{15} = \frac{1-s_{15}}{2}$
Baroni-Urbani and Buser (1976)	$s_{16} = \frac{a+\sqrt{ad}}{a+b+c+\sqrt{ad}}$	$u_{16} = 1 - s_{16}$
Yule (1927)	$s_{17} = \frac{ad-bc}{ad+bc}$	$u_{17} = \frac{1-s_{17}}{2}$
Yule (1912)	$s_{18} = \frac{\sqrt{ad}-\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}$	$u_{18} = \frac{1-s_{18}}{2}$
Sokal and Sneath (1963),un_4	$s_{19} = \frac{1}{4}(\frac{a}{a+b} + \frac{a}{a+c} + \frac{d}{d+b} + \frac{d}{d+c})$	$u_{19} = 1 - s_{19}$
Sokal and Sneath (1963), un_3		$u_{20} = \frac{b+c}{a+d}$
Gower & Legendre (1986)	$s_{21} = \frac{a+d}{a+\frac{(b+c)}{2}+d}$	$u_{21} = 1 - s_{21}$
Hamming distance		$u_{22} = \sum_{i=1}^{p}(x_i - y_i)^2$

We can find in [Batagelj and Bren, 1992] some relationships between these inequalities: $T7_{(Ultrametric)} \Rightarrow T6_{(Triangular)} \Leftarrow T8_{(Buneman)}$.

A dissimilarity measure which satisfies the properties T5 and T6 is a distance.

For binary data, we give in Table 2 some conventional proximity measures defined on $\{0,1\}^p$.

Let $x = (x_i)_{i=1,\ldots,p}$ and $y = (y_i)_{i=1,\ldots,p}$ two points in $\{0,1\}^p$ representing respectively attributes of two any objects x and y, we have: $a = \sum_{i=1}^{p} x_i y_i$ (resp. $d = \sum_{i=1}^{p}(1-x_i)(1-y_i)$ the cardinal of the subset of the attributes possessed in common (resp. not possessed by any of the two objects). $b = \sum_{i=1}^{p} x_i(1-y_i)$ (resp. $c = \sum_{i=1}^{p}(1-x_i)y_i$ the cardinal of the subset of the attributes possessed by the object x (resp. y) and not possessed by y (resp. x). Type 2 measures take in account also the cardinal d. The cardinals a, b, c and d are linked by the relation $a+b+c+d = p$.

3 Preorder Equivalence

3.1 Comparison between Two Proximity Indices

It is easy to see that on the same data set, two proximity measures u_i and u_j generally lead to different proximity matrices. But can we say that these two proximity measures are different? Many articles have been devoted to this issue. We can find in [Lerman, 1967] a proposal which says that two proximity measures u_i and u_j are equivalent if the preorders induced by each of the measures on all pairs of objects are identical. Hence the following definition.

Definition 1 (Equivalence in preordonnance:). let n objects $x,y,z...$ of $\mathbb{R}^p$ and any two proximity measures u_i and u_j on these objects. If for any quadruple (x,y,z,t), $u_i(x,y) \le u_i(z,t) \Rightarrow u_j(x,y) \le u_j(z,t)$ then, the two measures u_i and u_j are considered equivalent.

This definition was subsequently reproduced in many papers such as the following [Lesot et al., 2009], [Batagelj and Bren, 1995], [Bouchon-Meunier et al., 1996] and [Schneider and Borlund, 2007a] but the last one do not mention [Lerman, 1967]. This definition leads to an interesting theorem, the demonstration is in the reference [Batagelj and Bren, 1995].

Theorem 1 (Equivalence in preordonnance:). *let two proximity measures u_i and u_j, if there is a function f strictly monotone such that for every pair objects (x,y) we have: $u_i(x,y) = f(u_j(x,y))$, then u_i and u_j induce identical preorders and therefore they are equivalent: $u_i \equiv u_j$.*

The inverse is also true, ie, two proximity measures that depend on each other induce the same preorder and are, therefore, equivalent.

In order to compare proximity measures u_i and u_j, we need to define an index that could be used as a dissimilarity value between them. We denote this by $D(u_i,u_j)$. For example, we can use the following dissimilarity index which is based on preordonnance:

$$D(u_i,u_j) = \frac{1}{n^4}\sum_x\sum_y\sum_z\sum_t \delta_{ij}(x,y,z,t)$$

$$\text{where} \quad \delta_{ij}(x,y,z,t) = \begin{cases} 0 \text{ if } [u_i(x,y) - u_i(z,t)] \times [u_j(x,y) - u_j(z,t)] > 0 \\ \quad \text{or } u_i(x,y) = u_i(z,t) \text{ and } u_j(x,y) = u_j(z,t) \\ 1 \text{ otherwise} \end{cases}$$

D varies in the range $[0,1]$. Hence, for two proximity measures u_i and u_j, a value of 0 means that the preorder induced by the two proximity measures is the same and therefore the two proximity matrices of ui and uj are equivalent. The comparison between indices of proximity has been studied by [Schneider and Borlund, 2007a, Schneider and Borlund, 2007b] under a statistical perspective. The authors propose an empirical approach that aims to comparing proximity matrices obtained by each proximity measure on the pairs of objects. Then, they propose to test whether the matrices are statistically different or not using the Mantel test [Mantel, 1967]. In this work, we do not discuss the choice of comparison measure of proximity matrices. We simply use the expression presented above. Let specify again that our goal is not to compare proximity matrices or the preorders induced but to propose a different approach which is the topological equivalence that we compare to the preordering equivalence and we will put in perspective this two approaches.

With this proximity measure, we can compare proximity measures from their associated proximity matrices. The results of the comparison pair of proximity measures are given in Appendix Tables 3 and 4.

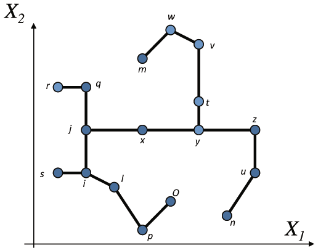
4 Topological Equivalence

The topological equivalence is in fact based on the concept of topological graph that use the neighborhood graph. The basic idea is quite simple: two proximity measures are equivalent if the topological graph induced on the set of objects is the same. For evaluating the resemblance between proximity measures, we compare neighborhood graphs and quantify their similarity. At first, we will define precisely what is a topological graph and how to build it. Then, we propose a proximity measure between topological graphs used to compare proximity measures in the section below.

4.1 Topological Graph

Let consider a set of objects $E = \{x,y,z,\ldots\}$ of $n = |E|$ objects in $\mathbb{R}^p$, such that $x,y,z,\ldots$ a set of points of $\mathbb{R}^p$. By using a proximity measure u, we can define a neighborhood relationship V_u to be a binary relation on $E \times E$. There are many possibilities to build a neighborhood binary relation.

For example, we can built the Minimal Spanning Tree (MST) on $(E \times E)$ and define, for two objects x and y, the property of the neighborhood according to minimal spanning tree [Kim and Lee, 2003], if they are directly connected by an edge. In this case, $V_u(x,y) = 1$ otherwise $V_u(x,y) = 0$. So, V_u forms the adjacency matrix associated to the MST graph, consisting of 0 and 1. Figure 1 shows a result in $\mathbb{R}^2$.



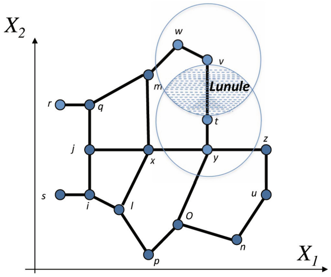
$$\left(\begin{array}{c|ccccccc}
 & \ldots & x & y & z & t & u & \ldots \\ \hline
\vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \\
x & \ldots & 1 & 1 & 0 & 0 & 0 & \ldots \\
y & \ldots & 1 & 1 & 1 & 1 & 0 & \ldots \\
z & \ldots & 0 & 1 & 1 & 0 & 1 & \ldots \\
t & \ldots & 0 & 1 & 0 & 1 & 0 & \ldots \\
u & \ldots & 0 & 0 & 1 & 0 & 1 & \ldots \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{array}\right)$$

Fig. 1 MST example for a set of points in $\mathbb{R}^2$ and the associated adjacency matrix

We can use many definitions to build the binary neighborhood, for example, the Graph Neighbors Relative (GNR), [Toussaint, 1980], [Preparata and Shamos, 1985], where all pairs of neighbor points (x,y) satisfy the following property:

$$\begin{array}{ll}
\text{if} & u(x,y) \leq \max(u(x,z),u(y,z)) \;\; ; \; \forall z \neq x, \neq y \\
\text{then,} & V_u(x,y) = 1 \;\; \text{otherwise} \;\; V_u(x,y) = 0.
\end{array}$$

Which geometrically means that the hyper-lunula (intersection of the two hyper-spheres centered on the two points) is empty. Figure 2 shows a result in $\mathbb{R}^2$. In this case, u is the Euclidean distance: $u_E(x,y) = \sqrt{(\sum_{i=1}^{p}(x_i - y_i)^2)}$.

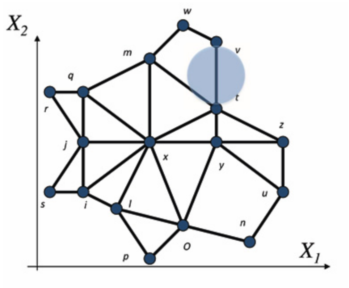


$$\left(\begin{array}{c|ccccccc}
 & \ldots & x & y & z & t & u & \ldots \\ \hline
\vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \\
x & \ldots & 1 & 1 & 0 & 0 & 0 & \ldots \\
y & \ldots & 1 & 1 & 1 & 1 & 0 & \ldots \\
z & \ldots & 0 & 1 & 1 & 0 & 1 & \ldots \\
t & \ldots & 0 & 1 & 0 & 1 & 0 & \ldots \\
u & \ldots & 0 & 0 & 1 & 0 & 1 & \ldots \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{array}\right)$$

Fig. 2 RNG example for a set of points in $\mathbb{R}^2$ and the associated adjacency matrix

Similarly, we can use the Gabriel Graph (GG), [Park et al., 2006], where all pairs of points satisfy: $u(x,y) \leq \min(\sqrt{u^2(x,z) + u^2(y,z)}) \;\; ; \; \forall z \neq x, \neq y$.

Geometrically, the diameter of the hypersphere $u(x,y)$ is empty. Figure 3 shows an example in $\mathbb{R}^2$.



$$\left(\begin{array}{c|ccccccc} & \dots & x & y & z & t & u & \dots \\ \hline \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \\ x & \dots & 1 & 1 & 0 & 1 & 0 & \dots \\ y & \dots & 1 & 1 & 1 & 1 & 0 & \dots \\ z & \dots & 0 & 1 & 1 & 1 & 1 & \dots \\ t & \dots & 1 & 1 & 1 & 1 & 0 & \dots \\ u & \dots & 0 & 0 & 1 & 0 & 1 & \dots \\ \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \end{array}\right)$$

Fig. 3 GG example for a set of points in $\mathbb{R}^2$ and the associated adjacency matrix

For a given neighborhood property (MST, GNR, GG), each measure u generates a topological structure on the objects E which is totaly described by its adjacency matrix V_u.

4.2 Comparing Adjacency Matrices

To fix ideas, let consider two proximity measures u_i and u_j taken among those we identified in Table 1 or in Table 2. $D_{u_i}(E \times E)$ and $D_{u_j}(E \times E)$ are the associated table of distances.

For a given neighborhood property, each of these two distances generates a topological structure on the objects E. A topological structure is fully described by its adjacency matrix. Note V_{u_i} and V_{u_j} the two adjacency matrices associated with two topological structures. To measure the degree of similarity between graphs, we only need to count the number of discordances between the two adjacency matrices. The matrix is symmetric, we can then calculate this amount by:

$$D(V_{u_i}, V_{u_j}) = \tfrac{2}{n(n-1)} \textstyle\sum_{k=1}^{n} \sum_{l=k+1}^{n} \delta_{kl} \quad \text{where} \quad \delta_{kl} = \begin{cases} 0 & \text{if } V_{u_i}(k,l) = V_{u_j}(k,l) \\ 1 & \text{otherwise} \end{cases}$$

D is the measure of dissimilarity which varies in the range $[0,1]$. Value 0 means that the two adjacency matrices are identical and therefore the topological structure induced by the two proximity measures is the same. In this case, we talk about topological equivalence between the two proximity measures. Value 1 means that the topology has changed completely, i.e., no pair of neighbors in the topological structure induced by the first proximity measure, only stayed close in the topological structure induced by the second measure and vice versa. D also interpreted as the percentage of disagreement between adjacency tables.

With this dissimilarity measure, we can compare proximity measures from their associated adjacency matrices. The results of pairwise comparisons of proximity measures are given in Appendix Tables 3 and 4.

5 Comparison and Discussion

To illustrate and compare the two approaches, we consider a relatively simple continuous and binary datasets, Fisher Iris and Zoo data from the UCI-Repository.

We will show some more general results. We deduce from the Theorem 1 of preordonnance equivalence, the following property.

Property. Let f be a strictly monotonic function of $\mathbb{R}^+$ in $\mathbb{R}^+$, u_i and u_j two proximity measures such as: $u_i(x,y) \to f(u_i(x,y)) = u_j(x,y)$ then,

$$u_i(x,y) \leq \max(u_i(x,z)\,,\, u_i(y,z)) \Leftrightarrow u_j(x,y) \leq \max(u_j(x,z)\,,\, u_j(y,z)).$$

Proof. Suppose: $\max(u_i(x,z)\,,\, u_i(y,z)) = u_i(x,z)$, by Theorem 1,

$$u_i(x,y) \leq u_i(x,z) \Rightarrow f(u_i(x,y)) \leq f(u_i(x,z)),$$

again, $$u_i(y,z) \leq u_i(x,z) \Rightarrow f(u_i(y,z)) \leq f(u_i(x,z))$$

$$\Rightarrow f(u_i(x,y)) \leq \max(f(u_i(x,z)), f(u_i(y,z))),$$

whence the result, $u_j(x,y) \leq \max(u_j(x,z), u_j(y,z))$.

The reciprocal implication is true, because f is continuous and strictly monotonic then its inverse f^{-1} is continuous in the same direction of variation of f.

In the case where f is strictly monotonic, we can say that if the preorder is preserved then the topology is preserved and vice versa. This property leads us to give the following theorem.

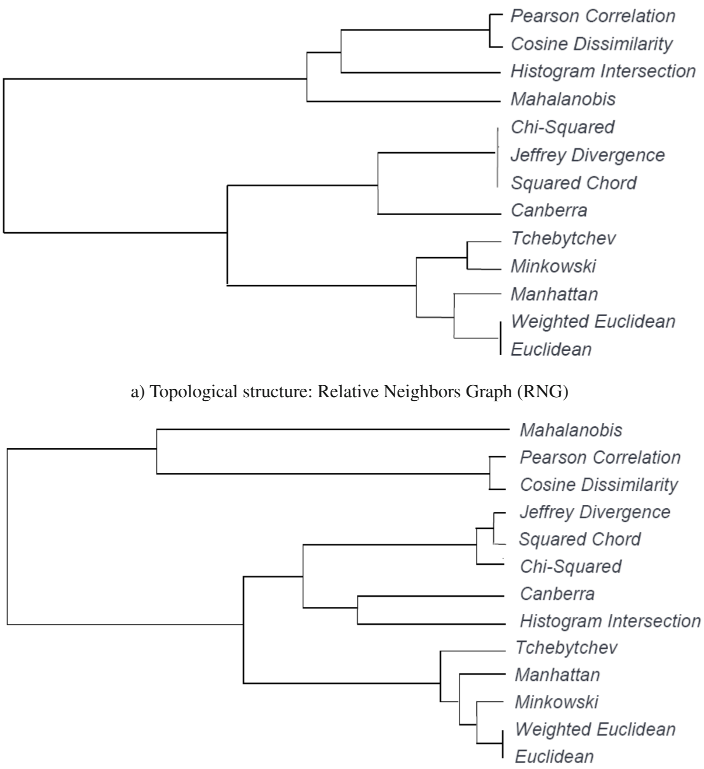
Theorem 2 (Equivalence in topology:). *Let u_i and u_j two proximity measures, if there exists a strictly monotonic f such that for every pair of objects (x,y) we have: $u_i(x,y) = f(u_j(x,y))$ then, u_i and u_j induce identical topological graphs and therefore they are equivalent: $u_i \equiv u_j$.*

The inverse is also true, ie two proximity measures which dependent on each other induce the same topology and are therefore equivalent.

Proposition. In the context of topological structures induced by the graph of neighbors relative, if two proximity measures u_i and u_j are equivalent in preordonnance, so they are necessarily topological equivalence.

Proof. If $u_i \equiv u_j$ (preordonnance equivalence) then,

$$u_i(x,y) \leq u_i(z,t) \Rightarrow u_j(x,y) \leq u_j(z,t) \;\; \forall x,y,z,t \in \mathbb{R}^p.$$



a) Topological structure: Relative Neighbors Graph (RNG)

b) Preordonnance

Fig. 4 Continuous data - Comparison of hierarchical trees

We have, especially for $t = x = y$ and $z \neq t$,

$$\begin{cases} u_i(x,y) \leq u_i(z,x) \Rightarrow u_j(x,y) \leq u_j(z,x) \\ u_i(x,y) \leq u_i(z,y) \Rightarrow u_j(x,y) \leq u_j(z,y) \end{cases}$$

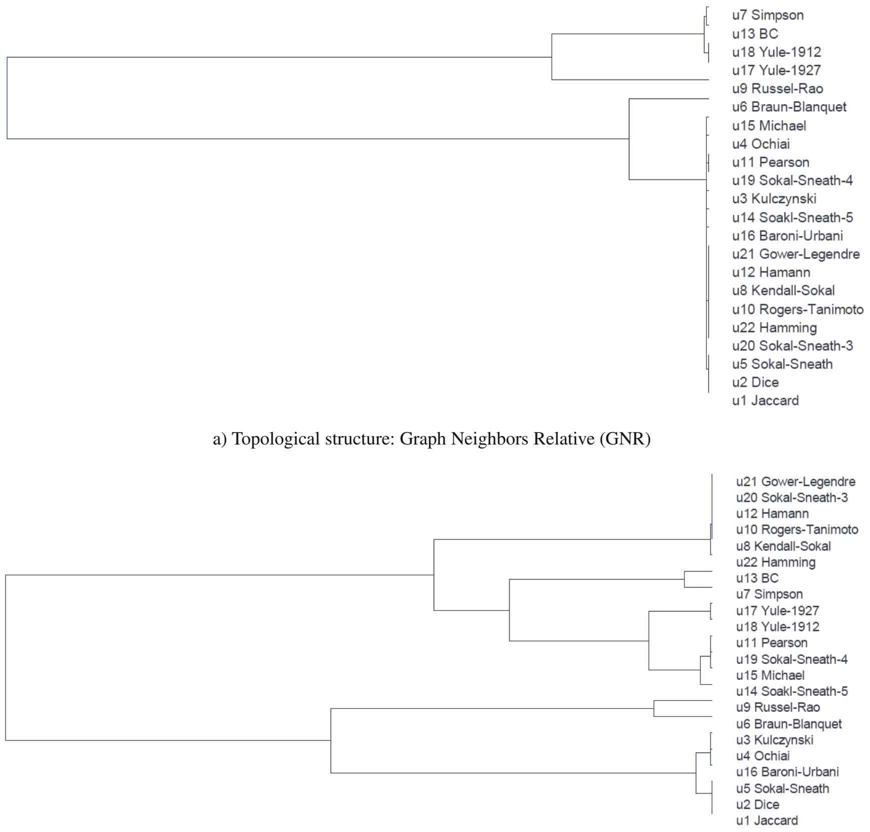
we deduce, $u_i(x,y) \leq \max(u_i(z,x), u_i(z,y)) \Rightarrow u_j(x,y) \leq \max(u_j(z,x), u_j(z,y))$ using symmetry property $P1$,

$$u_i(x,y) \leq \max(u_i(x,z), u_i(y,z)) \Rightarrow u_j(x,y) \leq \max(u_j(x,z), u_j(y,z))$$

hence, $u_i \equiv u_j$ (topological equivalence). □

Remark. Influence of structure: $u_i \equiv u_j$ (preordonnance equivalence) $\Rightarrow u_i \equiv u_j$ (GNR topological equivalence) $\Leftarrow u_i \equiv u_j$ (GG topological equivalence).

The results of pairwise comparisons, Appendix Table 3, are somewhat different, some are closer than others. We can note that three pairs of proximity measures (u_E, u_{E_w}), (u_{SC}, u_{JD}) and (u_{χ^2}, u_{JD}) which are in perfect preordonnance equivalence $(D(u_i, u_j) = 0)$ are in perfect topology equivalence $(D(V_{u_i}, Vu_j) = 0)$. But the inverse is not true, for example, the pair (u_{SC}, u_{χ^2}) which is in perfect topology equivalence is not in perfect preordonnance equivalence.



a) Topological structure: Graph Neighbors Relative (GNR)

b) Preordonnance

Fig. 5 Binary data - Comparison of Hierarchical trees

We can also see, Appendix Table 4, that the results of pairwise comparisons for binary data are not very different. All pairs which are in perfect preordonnance equivalence are in perfect topology equivalence. The pair (u_{14} Sokal-Sneath, u_{16} Baroni-Urbani) which is in perfect topology equivalence is not in perfect preordonnance equivalence.

To view these proximity measures, we propose, for example, to apply an algorithm to construct a hierarchy according to Ward's criterion [Ward Jr, 1963]. Proximity measures are grouped according to their degree of resemblance and they also compare their associated adjacency matrices. This yields the dendrograms below, Figures 4 and 5.

We found also that the classification results differ depending on comparing the proximity measures using preordonnance equivalence or topological equivalence.

6 Conclusion

The choice of a proximity measure is subjective because it depends often of habits or criteria such as the subsequent interpretation of results. This work proposes a new approach of equivalence between proximity measures. This approach, called topological, is based on the concept of neighborhood graph induced by the proximity measure. For the practical matter, in this paper the measures that we have compared, are built on continuous and binary data.

In our next work, we will apply a statistical test on the adjacency matrices associated to proximity measures because it helps to give a statistical significance of the degree of equivalence between two proximity measures and validates the topological equivalence, which means here, if they really induce the same neighborhood structure on the objects. In addition, we want to extend this work to other topological structures in order to analyze the influence of the choice of neighborhood structure on the topological equivalence between these proximity measures. Also, we want to analyze the influence of data and the choice of clustering methods on the regroupment of these proximity measures.

References

[Batagelj and Bren, 1992] Batagelj, V., Bren, M.: Comparing resemblance measures. Technical report, Proc. International Meeting on Distance Analysis, DISTANCIA 1992 (1992)

[Batagelj and Bren, 1995] Batagelj, V., Bren, M.: Comparing resemblance measures. Journal of Classification 12, 73–90 (1995)

[Bouchon-Meunier et al., 1996] Bouchon-Meunier, B., Rifqi, M., Bothorel, S.: Towards general measures of comparison of objects. Fuzzy Sets and Systems 84(2), 143–153 (1996)

[Clarke et al., 2006] Clarke, K., Somerfield, P., Chapman, M.: On resemblance measures for ecological studies, including taxonomic dissimilarities and a zero-adjusted bray-curtis coefficient for denuded assemblages. Journal of Experimental Marine Biology and Ecology 330(1), 55–80 (2006)

[Fagin et al., 2003] Fagin, R., Kumar, R., Sivakumar, D.: Comparing top k lists. In: Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, p. 36. Society for Industrial and Applied Mathematics (2003)

[Kim and Lee, 2003] Kim, J., Lee, S.: Tail bound for the minimal spanning tree of a complete graph. Statistics and Probability Letters 64(4), 425–430 (2003)

[Lerman, 1967] Lerman, I.: Indice de similarité et préordonnance associée, Ordres. Travaux du séminaire sur les ordres totaux finis, Aix-en-Provence (1967)

[Lesot et al., 2009] Lesot, M.-J., Rifqi, M., Benhadda, H.: Similarity measures for binary and numerical data: a survey. IJKESDP 1(1), 63–84 (2009)

[Liu et al.,] Liu, H., Song, D., Rüger, S.M., Hu, R., Uren, V.S.: Comparing Dissimilarity Measures for Content-Based Image Retrieval. In: Li, H., Liu, T., Ma, W.-Y., Sakai, T., Wong, K.-F., Zhou, G. (eds.) AIRS 2008. LNCS, vol. 4993, pp. 44–50. Springer, Heidelberg (2008)

[Malerba et al., 2002] Malerba, D., Esposito, F., Monopoli, M.: Comparing dissimilarity measures for probabilistic symbolic objects. Series Management Information Systems 6, 31–40 (2002)

[Mantel, 1967] Mantel, N.: A technique of disease clustering and a generalized regression approach. Cancer Research 27, 209–220 (1967)

[Noreault et al., 1980] Noreault, T., McGill, M., Koll, M.: A performance evaluation of similarity measures, document term weighting schemes and representations in a boolean environment. In: Proceedings of the 3rd Annual ACM Conference on Research and Development in Information Retrieval, p. 76. Butterworth and Co. (1980)

[Park et al., 2006] Park, J., Shin, H., Choi, B.: Elliptic gabriel graph for finding neighbors in a point set and its application to normal vector estimation. Computer-Aided Design 38(6), 619–626 (2006)

[Preparata and Shamos, 1985] Preparata, F., Shamos, M.: Computational geometry: an introduction. Springer (1985)

[Richter, 1992] Richter, M.: Classification and learning of similarity measures. In: Proceedings der Jahrestagung der Gesellschaft fur Klassifikation. Studies in Classification, Data Analysis and Knowledge Organisation. Springer (1992)

[Schneider and Borlund, 2007a] Schneider, J., Borlund, P.: Matrix comparison, part 1: Motivation and important issues for measuring the resemblance between proximity measures or ordination results. Journal American Society for Information Science and Technology 58(11), 1586–1595 (2007a)

[Schneider and Borlund, 2007b] Schneider, J., Borlund, P.: Matrix comparison, part 2: Measuring the resemblance between proximity measures or ordination results by use of the mantel and procrustes statistics. Journal American Society for Information Science and Technology 58(11), 1596–1609 (2007b)

[Spertus et al., 2005] Spertus, E., Sahami, M., Buyukkokten, O.: Evaluating similarity measures: a large-scale study in the orkut social network. In: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, p. 684. ACM (2005)

[Strehl et al., 2000] Strehl, A., Ghosh, J., Mooney, R.: Impact of similarity measures on webpage clustering. In: Workshop on Artificial Intelligence for Web Search (AAAI 2000), pp. 58–64 (2000)

[Toussaint, 1980] Toussaint, G.: The relative neighbourhood graph of a finite planar set. Pattern Recognition 12(4), 261–268 (1980)

[Ward Jr, 1963] Ward Jr., J.: Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association 58(301), 236–244 (1963)

[Warrens, 2008] Warrens, M.: Bounds of resemblance measures for binary (presence/absence) variables. Journal of Classification 25(2), 195–208 (2008)

[Zhang and Srihari, 2003] Zhang, B., Srihari, S.: Properties of binary vector dissimilarity measures. In: Proc. JCIS Int'l Conf. Computer Vision, Pattern Recognition, and Image Processing. Citeseer (2003)

[Zwick et al., 1987] Zwick, R., Carlstein, E., Budescu, D.: Measures of similarity among fuzzy concepts: A comparative analysis. Int. J. Approx. Reason. 1(2), 221–242 (1987)

Appendix

Table 3 Similarities tables: $S(V_{u_i}, V_{u_j}) = 1 - D(V_{u_i}, V_{u_j})$ and $S(u_i, u_j) = 1 - D(u_i, u_j)$. Continuous data - Topology (row) & Preordonnance (column).

$S = 1 - D$	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}
$u_1 : u_E$	**1**	.776	.973	.988	.967	.869	.890	.942	**1**	.947	.945	.926	.863
$u_2 : u_{Mah}$	.876	**1**	.773	.774	.752	.701	.707	.737	.776	.739	.738	.742	.703
$u_3 : u_{Man}$	.964	.840	**1**	.964	.940	.855	.882	.930	.973	.933	.932	.924	.848
$u_4 : u_{Min\gamma}$	.964	.876	.947	**1**	.967	.871	.892	.946	.988	.950	.949	.925	.866
$u_5 : u_{Tch}$	.947	.858	.929	.964	**1**	.865	.887	.940	.957	.942	.942	.914	.860
$u_6 : u_{Cos}$	.858	.858	.840	.840	.858	**1**	.893	.898	.869	.899	.899	.830	.957
$u_7 : u_{Can}$	.911	.840	.929	.893	.911	.822	**1**	.943	.890	.940	.942	.874	.868
$u_8 : u_{SC}$	.947	.840	.947	.929	.947	.858	.947	**1**	.942	.957	**1**	.913	.884
$u_9 : u_{Ew}$	**1**	.876	.964	.964	.947	.858	.911	.947	**1**	.947	.945	.926	.863
$u_{10} : u_{\chi^2}$	.947	.840	.947	.929	.947	.858	.947	**1**	.947	**1**	**1**	.912	.885
$u_{11} : u_{JD}$	.947	.840	.947	.929	.947	.858	.947	**1**	.947	**1**	**1**	.914	.884
$u_{12} : u_{HIM}$	.884	.813	.884	.867	.902	.884	.884	.920	.884	.920	.920	**1**	.825
$u_{13} : u_\rho$	.867	.849	.831	.867	.867	.973	.796	.849	.867	.849	.849	.876	**1**

The elements located above the main diagonal correspond to the dissimilarities in preordonnance and those below correspond to the dissimilarities in topology.

Table 4 Similarities tables: $S(u_i,u_j)=1-D(u_i,u_j)$ and $S(V_{u_i},V_{u_j})=1-D(V_{u_i},V_{u_j})$. Binary data – Preordonnance (row) & Topology (column).

S = 1 - D	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}	u_{15}	u_{16}	u_{17}	u_{18}	u_{19}	u_{20}	u_{21}	u_{22}
u_1: Jaccard	**1**	**1**	.994	.990	**1**	.941	.908	.987	.838	.987	.992	.987	.909	.996	.982	.998	.922	.922	.992	.987	.987	.987
u_2: Dice	**1**	**1**	.994	.990	**1**	.941	.908	.987	.838	.987	.992	.987	.909	.996	.982	.998	.922	.922	.992	.987	.987	.987
u_3: Kulczynski	.964	.964	**1**	.987	.994	.935	.914	.988	.838	.988	.998	.988	.914	.997	.987	.996	.928	.928	.998	.988	.988	.988
u_4: Ochiai	.975	.975	.984	**1**	.990	.930	.901	.980	.828	.980	.985	.980	.902	.989	.974	.991	.915	.915	.985	.980	.980	.980
u_5: Sokal & Sneath	**1**	**1**	.964	.975	**1**	.941	.908	.987	.838	.987	.992	.987	.909	.996	.982	.998	.922	.922	.992	.987	.987	.987
u_6: Braun & Blanquet	.923	.922	.899	.910	.922	**1**	.850	.939	.875	.939	.933	.939	.851	.937	.924	.939	.865	.865	.933	.939	.939	.939
u_7: Simpson	.831	.831	.866	.852	.831	.766	**1**	.910	.906	.910	.916	.910	.977	.912	.909	.910	.986	.986	.916	.910	.910	.910
u_8: Kendall & Sokal	.855	.855	.865	.855	.855	.787	.816	**1**	.832	**1**	.989	**1**	.910	.988	.977	.989	.919	.919	.989	**1**	**1**	**1**
u_9: Russel & Rao	.852	.852	.821	.833	.852	.893	.759	.711	**1**	.832	.838	.832	.886	.836	.834	.837	.900	.900	.838	.832	.832	.832
u_{10}: Rogers & Tanimoto	.855	.855	.865	.855	.855	.787	.816	**1**	.711	**1**	.989	**1**	.910	.988	.977	.989	.919	.919	.989	**1**	**1**	**1**
u_{11}: Pearson	.899	.899	.933	.920	.899	.838	.872	.917	.756	.917	**1**	.989	.917	.996	.986	.994	.930	.930	**1**	.989	.989	.989
u_{12}: Hamann	.855	.855	.865	.855	.855	.786	.816	**1**	.711	**1**	.917	**1**	.910	.988	.977	.989	.919	.919	.989	**1**	**1**	**1**
u_{13}: BC	.779	.779	.813	.799	.779	.717	.860	.869	.646	.869	.878	.869	**1**	.913	.910	.910	.986	.986	.917	.910	.910	.910
u_{14}: Sokal & Sneath 5	.932	.932	.963	.951	.932	.870	.867	.899	.788	.899	.967	.899	.845	**1**	.986	**1**	.926	.926	.996	.988	.988	.988
u_{15}: Michael	.899	.899	.931	.921	.899	.838	.864	.908	.764	.908	.981	.908	.869	.962	**1**	.983	.923	.923	.986	.977	.977	.977
u_{16}: Baroni & Urbani	.972	.972	.965	.970	.972	.901	.845	.883	.827	.883	.927	.883	.806	.959	.923	**1**	.924	.924	.994	.989	.989	.989
u_{17}: Yule 1927	.857	.857	.891	.877	.857	.795	.921	.876	.723	.876	.947	.876	.920	.924	.930	.884	**1**	**1**	.930	.919	.919	.919
u_{18}: Yule 1912	.857	.857	.891	.877	.857	.795	.922	.876	.724	.876	.947	.876	.920	.924	.930	.884	**1**	**1**	.930	.919	.919	.919
u_{19}: Sokal & Sneath 4	.899	.899	.933	.919	.899	.837	.873	.916	.755	.916	**1**	.916	.877	.967	.980	.927	.947	.947	**1**	.989	.989	.989
u_{20}: Sokal & Sneath 3	.855	.855	.865	.855	.855	.787	.816	**1**	.711	**1**	.917	**1**	.869	.899	.908	.883	.876	.876	.916	**1**	**1**	**1**
u_{21}: Gower & Legendre	.855	.855	.865	.855	.855	.787	.816	**1**	.711	**1**	.917	**1**	.869	.899	.908	.883	.876	.876	.916	**1**	**1**	**1**
u_{22}: Hamming distance	.855	.855	.865	.855	.855	.787	.816	**1**	.711	**1**	.917	**1**	.869	.899	.908	.883	.876	.876	.916	**1**	**1**	**1**

Comparing Two Discriminant Probabilistic Interestingness Measures for Association Rules

Israël César Lerman and Sylvie Guillaume

Abstract. Preliminary nomalization is needed for probabilistic pairwise comparison between attributes in Data Mining. Normalization plays a very important part in preserving the discriminant property of the probability scale when the number of observations becomes large. Asymmetrical associations between boolean attributes are considered in our paper. Its goal consists of comparison between two approaches. The first one is due to a normalized version of the "Likelihood Linkage Analysis" methodology. The second one is based on the notion of "Test Value" defined with respect to a hypothetical sample, sized 100 and summarizing the initial observed sample. Two facets are developed in our work: theoretical and experimental. A comparative experimental analysis is presented with the well known databases "Wages" and "Abalone".

1 Introduction

By considering a boolean description in the context of a given database, a fundamental and well known problem in *Data Mining* consists in discovering a significant and reduced set of association rules between parts of the boolean description. For simplicity, but without loss of generality, let us assume a set $\mathcal{A}$ of boolean attributes describing a set $\mathcal{O}$ of objects. Denote by n and p the respective cardinalities of $\mathcal{O}$ and $\mathcal{A}$. The set $\mathcal{O}$ is generally provided from a universe $\mathcal{U}$ of objects. On the other hand, the boolean attribute set $\mathcal{A}$ can be obtained from conjunctions called itemsets of more elementary boolean attributes.

Israël César Lerman
IRISA - Université de Rennes 1, Campus de Beaulieu, 35042 Rennes Cedex
e-mail: lerman@irisa.fr

Sylvie Guillaume
Clermont Université, Auvergne, LIMOS, BP 10448, F-63000 Clermont-Fd
e-mail: sylvie.guillaume@udamail.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 59–83.
DOI: 10.1007/978-3-642-35855-5_4

Intuitively, for a given ordered pair $(a,b) \in \mathcal{A} \times \mathcal{A}$ an association rule written as $a \rightarrow b$ means that generally but not absolutely, a $TRUE$ value of a on a given element o of $\mathcal{O}$ implies a $TRUE$ value of b on o. In order to discover and to evaluate such a directed implicative tendency, a relevant statistical association measure is needed. The latter is expected to distinguish the most "interesting" rules, those which increase our "knowledge" in the network of the associative tendencies between the observed boolean descriptions. In the seminal work [1] where the famous *Confidence* and *Support* coefficients were proposed, requirements of rule quality measurements are not considered. Nevertheless, such requirements are clearly expressed in [28] where several criteria are studied in order to categorize a set of statistical coefficients defining measures of interestingness for association rules. New aspects are also considered in comparative studies [11, 12]. The notion of statistical independence between descriptive attributes appears as an important part in the development of many of the proposed coefficients [11, 12, 20, 24, 28].

Initially, a probability scale has been setup with respect to a probabilistic independence hypothesis in order to validate the dependency between the two components a and b of a single pair (a,b) of descriptive attributes.

The idea of the *Likelihood Linkage Analysis* methodology [4, 6, 13, 14, 15, 18, 20, 21], is not to test the existence of a real link or even a given dependency level between two single attributes a and b, but to use the probability scale in order to mutually compare a set of attribute pairs from $\mathcal{A}$. For this pairwise comparison, the adaptation of such a probability scale requires a preliminary normalization. This plays a crucial part in preserving the discriminant property of the probability scale when the number of observations n ($n = card(\mathcal{O})$) is very large (the order of magnitude of n can reach many millions). To be precise, this discriminant property when measuring by a probability scale allows a reduced and manageable set of "interesting" association rules to be selected.

The goal of this paper consists of comparing two normalization approaches leading to two types of indices. The first one results from the "*Likelihood of the Relational Links Analysis*" (see the previous references). It has a *contextual* nature with respect to a "relevant" set of association rules. It leads to the notion of "*Contextual Likelihood of the Link Implication*" [20]. The second approach has been proposed more recently [23, 26]. It refers in its principle more directly to "statistical tests of independence hypotheses". In this approach data is summarized by means of a 100 sized sample synthetizing in some way, the initial data set and a *Test Value* index denoted $TV100$ proposed on the basis of a reduced sample. However, the basic notion of a statistical data unit is no longer respected. New interpretations and new alternatives to this type of approach are considered in our paper.

In section 2 we will compare conceptually and before any normalization the two approaches *LL* ("*Likelihood of the Link*", "*Vraisemblance du Lien*") and *TV* ("*Test Value*"). In Section 3 we will begin by recalling the expression of the normalized version of the Likelihood of the Link index. This index is also called "*Contextual Likelihood of the Link Implication*" or "*Contextual Intensity of Implication*" [20]. Next, we will analyse the $TV100$ approach or more generally, that TVe where a synthetic reduction of the initial sample size n to e, is proposed. This analysis will lead us to set up three different versions of this type of approach. Two of them are new. The comparison between the two approaches *normalized LL* and TVe includes two facets: theoretical and experimental. For the purposes of this comparison, increasing models of the number n of observations are considered. For this, two increasing models denoted by M_1 and M_2 will be defined in Section 4.1. M_1 is a classical model. The second model M_2 is more recent and is a specific one. It has been suggested by Yves Kodratoff (personal communication studied in [20]). The theoretical comparison of both types of indices will be considered in Section 4.2. The experimental analysis of the behaviour of the four indices - normalized LL and three versions of TVe (for $e = 100$) will be reported in Section 5. Section 6 will end with concluding remarks and some prospects.

2 *LL* and *TV*

For a given ordered pair (a,b) of boolean attributes belonging to $\mathcal{A} \times \mathcal{A}$ (see Section 1) let us introduce the conjunctions $a \wedge b$, $a \wedge \bar{b}$, $\bar{a} \wedge b$ and $\bar{a} \wedge \bar{b}$ where $\bar{a}$ (resp., $\bar{b}$) indicates the negated attribute of a (resp., b). The respective set theoretic representations of these conjunctions are $\mathcal{O}(a \wedge b) = \mathcal{O}(a) \cap \mathcal{O}(b)$, $\mathcal{O}(a \wedge \bar{b}) = \mathcal{O}(a) \cap \mathcal{O}(\bar{b})$, $\mathcal{O}(\bar{a} \wedge b) = \mathcal{O}(\bar{a}) \cap \mathcal{O}(b)$ and $\mathcal{O}(\bar{a} \wedge \bar{b}) = \mathcal{O}(\bar{a}) \cap \mathcal{O}(\bar{b})$, where for a given attribute c describing $\mathcal{O}$, $\mathcal{O}(c)$ denotes the $\mathcal{O}$ subset where c is $TRUE$. The cardinalities of these subsets will be designated by $n(a \wedge b)$, $n(a \wedge \bar{b})$, $n(\bar{a} \wedge b)$ and $n(\bar{a} \wedge \bar{b})$, respectively. These cardinalities define the entries of the contingency table crossing the two binary attributes $\{a,\bar{a}\}$ and $\{b,\bar{b}\}$, see table 2 in Section 3. By dividing these cardinalities by n, the relative frequencies or proportions $p(a \wedge b)$, $p(a \wedge \bar{b})$, $p(\bar{a} \wedge b)$ and $p(\bar{a} \wedge \bar{b})$ are obtained, respectively. Obviously, the sum of these proportions is equal to 1.

2.1 The LL (Likelihood of the Link, **Vraisemblance du Lien***) Approach*

The symmetrical comparison between two boolean attributes a and b [4, 13, 14, 15, 18] has been addressed before the asymmetrical one [6, 20, 21]. In the latter case the implicative tendency $a \rightarrow b$ has to be established and evaluated. Let us now recall the general scheme set up for the symmetrical case. The first step consists in introducing a “raw” index $s(a,b)$ representing the number of objects where both attributes have a $TRUE$ value. More precisely, $s(a,b) = n(a \wedge b) = card[\mathcal{O}(a) \cap \mathcal{O}(b)]$. A probabilistic hypothesis denoted $\mathcal{N}$, of no relation or independence between a and b is introduced in order to evaluate how large $s(a,b)$ is with respect to the sizes of $n(a) = card[\mathcal{O}(a)]$ and $n(b) = card[\mathcal{O}(b)]$. With the observed attribute pair (a,b) a random pair denoted by $(a^\star, b^\star)$ is associated with the observed attribute pair. $a^\star$ and $b^\star$ are two independent random attributes defined in the context of a random model specified in such a way that the mathematical expectations of $n(a^\star) = card[\mathcal{O}(a^\star)]$

and $n(b^\star) = card[\mathcal{O}(b^\star)]$ are equal to $n(a)$ and $n(b)$, respectively [15, 20]. Under these conditions, the symmetrical version of the *Likelihood of the Link* index can be written

$$P(a,b) = Pr^{\mathcal{N}}\{n(a^\star \wedge b^\star) < n(a \wedge b)\} \tag{1}$$

where $n(a^\star \wedge b^\star) = card[\mathcal{O}(a^\star) \cap \mathcal{O}(b^\star)]$

An adapted version of this type of measure for the asymmetrical implicative case has been introduced by [6]. Instead of evaluating the unlikelihood of the "bigness" of $n(a \wedge b)$, the evaluation concerns the "smallness" of $n(a \wedge \bar{b})$. Under these conditions, the general form of the Likelihood of the Link probabilistic index for the directed association $a \rightarrow b$ can be expressed by

$$\mathcal{I}(a \rightarrow b) = Pr^{\mathcal{N}}\{n(a^\star \wedge \bar{b^\star}) > n(a \wedge \bar{b})\} \tag{2}$$

It has been called by R. Gras "*Intensity of Implication*" [7].

The Poisson model has proved to be the most suitable one for defining such an association. For this model and under the probabilistic hypothesis of no relation (independence) $n(a^\star \wedge \bar{b^\star})$ follows a Poisson probability law parametrized by $n(a) \times n(\bar{b})/n$ [14, 15, 20, 21]. Let us recall here this model defined in the previous references. For this purpose consider a triple $(\mathcal{O}; E, F)$ where E and F are two subsets of the object set $\mathcal{O}$. Denote by c and d the respective cardinalities of E and F: $c = card(E)$ and $d = card(F)$. In our context $E = \mathcal{O}(a)$ and $F = \mathcal{O}(\bar{b})$, $c = n(a)$ and $d = n(\bar{b})$. The random model makes correspondence between the triple $(\mathcal{O}; E, F)$ and a random one $(\mathcal{O}^\star; E^\star, F^\star)$ where $\mathcal{O}^\star$, $E^\star$ and $F^\star$ are three random sets defined such that, for a given value $\mathcal{O}_1$ of $\mathcal{O}^\star$, $E^\star$ and $F^\star$ are two independent random subsets of $\mathcal{O}_1$. The only requirement for the random set $\mathcal{O}^\star$ concerns its cardinality denoted $n^\star$: $n^\star$ follows a Poisson probability law parametrized by $n = card(\mathcal{O})$. Thus, for a given positive integer m ($m \geq 0$), $Pr\{n^\star = m\} = \frac{n^m}{m!} \times e^{-n}$.

Now, let us precise how the random subset $E^\star$ is chosen in $\mathcal{O}_1$. Choosing $F^\star$ in $\mathcal{O}_1$ is similar. Denote m the cardinality of $\mathcal{O}_1$ and introduce the set $\mathcal{P}(\mathcal{O}_1)$ of all subsets of $\mathcal{O}_1$. Consider now the inclusion relation between subsets of $\mathcal{O}_1$. This relation defines a lattice structure on $\mathcal{P}(\mathcal{O}_1)$. It comprises $m+1$ levels. The k^{th} level is constituted by all $\mathcal{O}_1$ subsets whose cardinality is k, $0 \leq k \leq m$. Two steps are required for choosing $E^\star$. The first one consists in randomly choosing a level k. This is performed with the binomial probability $\binom{m}{k} p^k (1-p)^{m-k}$, where p is defined by the proportion $\frac{c}{n}$ ($c = card(E)$). For a given chosen level, $E^\star$ is randomly taken on this level, provided by a uniform probability distribution (equal chance to be chosen for each subset of the concerned level). In [15, 21] we establish that $card(E^\star \cap F^\star)$ follows a Poisson distribution law parametrized by $c \times d/n$. The latter is the mathematical expectation of this probability distribution. Under these conditions the right member of Equation 2 is given by

$$\sum_{l=n(a \wedge \bar{b})+1}^{\infty} \frac{\lambda^l}{l!} \times e^{-\lambda}$$

where $\lambda = (n(a) \times n(b))/n$.

Let us introduce here the following standardized version of $n(a \wedge \bar{b})$ with respect to the mean and standard deviation of the associated random index $n(a^\star \wedge \bar{b^\star})$

$$Q(a,\bar{b}) = \frac{n(a \wedge \bar{b}) - [n(a) \times n(\bar{b})/n]}{\{n(a) \times n(\bar{b})/n\}^{\frac{1}{2}}} \tag{3}$$

By using the Poisson probability distribution with parameter $n(a) \times n(\bar{b})/n$, $\mathcal{I}(a \rightarrow b)$ (see Equation (2)) can be exactly computed even for large values of n. However, for n large enough and $n(a) \times n(\bar{b})/n$ not too small, the Normal distribution provides a very accurate approximation for the concerned Poisson probability distribution [5]. This leads to the following analytical equation

$$Pr^{\mathcal{N}}\{n(a^\star \wedge \bar{b^\star}) > n(a \wedge \bar{b})\} = Pr\{Q(a^\star, \bar{b^\star}) > Q(a,\bar{b})\} \simeq \Phi[-Q(a,\bar{b})] \tag{4}$$

where Φ designates the standard normal cumulative distribution function. In real databases the above mentioned conditions (n large enough and $\frac{n(a) \times n(\bar{b})}{n}$ not too small) are satisfied. Therefore, $\mathcal{I}(a \rightarrow b)$ and $\Phi[-Q(a,\bar{b})]$ can be considered as identical.

2.2 *The* TV *(Test Value,* Valeur Test*) Approach*

In the spirit of the *LL* approach, the null hypothesis of no relation between attributes, denoted above by $\mathcal{N}$, has to be rejected. However, $\mathcal{N}$ is crucially important to establish a probability scale for mutually comparing the links between attributes [4, 16]. Otherwise, conceptually, the TV approach is more closely related to the theory of independence hypothesis testing. Then, the *p-value* critical threshold has an essential part in defining the *Test Value*. This threshold becomes here

$$p = Pr^{\mathcal{N}}\{n(a^\star \wedge \bar{b}^\star) \leq n(a \wedge \bar{b})\} \tag{5}$$

$p < 0.5$ in the case of an implicative tendency $a \rightarrow b$. By comparing with Equation (2), we have $\mathcal{I}(a \rightarrow b) = 1 - p$. The *Test Value* of $a \rightarrow b$ denoted by $TV(a \rightarrow b)$ is defined by

$$TV(a \rightarrow b) = \Phi^{-1}(1-p) \tag{6}$$

Equation (6) is equivalent to

$$\Phi[TV(a \rightarrow b)] = 1 - p = \mathcal{I}(a \rightarrow b) \tag{7}$$

Now, reconsider from (4) the identification between $\Phi[-Q(a,\bar{b})]$ and $\mathcal{I}(a \rightarrow b)$. This entails that $\Phi[-Q(a,\bar{b})] = \Phi[TV(a \rightarrow b)]$. By taking into account the strict increasing property of the function Φ, we obtain the following important result

Proposition 1. $TV(a \rightarrow b) = -Q(a,\bar{b})$

Let us recall that $Q(a,\bar{b})$ may be expressed as a correlation coefficient between the two attributes a and $\bar{b}$ multiplied by $\sqrt{n}$ [20, 21]. The correlation coefficient value is determined from the proportions $p(a \wedge b)$, $p(a \wedge \bar{b})$, $p(\bar{a} \wedge b)$ and $p(\bar{a} \wedge \bar{b})$, defined above in Section 2. This value is positive or negative whether $p(a \wedge b) > p(a) \times p(b)$ or $p(a \wedge b) < p(a) \times p(b)$. Consequently, the $p - value$ (resp., $\mathcal{I}(a \rightarrow b)$) tends "very quickly" towards 0 or 1 (resp., 1 or 0) according to the first (resp., second) alternative. In table 1 the proportions $p(a \wedge b)$, $p(a \wedge \bar{b})$, $p(\bar{a} \wedge b)$ and $p(\bar{a} \wedge \bar{b})$ are constant. The first row of this table corresponds to the crossing between two binary attributes considered in [27] and taken up again in [23].The second row is deduced from the first one by multiplying the entries of the associated contingency table by 3. The third (resp., fourth) row is deduced from the second (resp., third) one by using the multiplicative factor 10 (resp., 2). Thus, the probability scale based on LL (resp., TV) approach is not able to discriminate the relationships between observed attributes on large object sets.

Table 1 Increasing behaviour of $\mathcal{I}(a \rightarrow b)$

n	$n(a)$	$n(\bar{b})$	$n(a \wedge \bar{b})$	$-Q(a,\bar{b})$	$\mathcal{I}(a \rightarrow b)$
91	25	51	9	1.339	0.873
273	75	153	27	2.319	0.990
2730	750	1530	270	7.332	1.000
5460	1500	3060	540	10.370	1.000

3 The *LL* Normalized Index and the *TVe* Indices

3.1 The LL Normalized Index VLgrImpP

The principle of the global reduction of association rule measures defined from a given observation set, has already been proposed in [21]. It has been taken up again and studied in-depth in [20]. In the context of a given database, let us consider a set $\mathcal{A}$ of boolean attributes on which an index of implication (association rule measure) has to be established. Denote $\mathcal{A} = \{a^j \mid 1 \leq j \leq p\}$ and introduce the cartesian product $\mathcal{A} \times \mathcal{A}$ defined by all ordered pairs of attributes from $\mathcal{A}$. Now, let us distinguish in $\mathcal{A} \times \mathcal{A}$ a subset $\mathcal{C}$ of attribute ordered pairs (a,b) for which the implication $a \rightarrow b$ "may have some meaning". Global reduction will be performed with respect to $\mathcal{C}$. A first condition required by an ordered pair of attributes (a,b) to belong to $\mathcal{C}$, is $n(a \wedge \bar{b}) < n(a) \times n(\bar{b})/n$. Besides, this condition is equivalent to $n(a \wedge b) > n(a) \times n(b)/n$. A further condition which can be requested is $n(a) < n(b)$. Indeed, in the case where the logical implication $a \rightarrow b$ is observed without counter-examples, one has $\mathcal{O}(a) \subset \mathcal{O}(b)$, where $\mathcal{O}(a)$ [resp., $\mathcal{O}(b)$] is the subset of objects where the attribute a (resp., b) is $TRUE$. Otherwise, in the case where $n(a) < n(b)$, we prove that $Q(a,\bar{b}) < Q(b,\bar{a})$ [22]. In [20] additional conditions defined by minimum values of the *Support* and *Confidence* indices are considered for (a,b) in order to belong to $\mathcal{C}$.

The simplest version of the $\mathcal{C}$ set of ordered attribute pairs is defined by

$$\mathcal{C}_0 = \left\{ (a,b) \mid (a,b) \in \mathcal{A} \times \mathcal{A}, n(a \wedge b) > \frac{n(a) \times n(b)}{n} \text{ and } n(a) < n(b) \right\}$$

The global reduction with respect to $\mathcal{C}_0$ is called a *complete reduction*. It will be taken into account in this paper. Therefore, let us consider the empirical distribution of $Q(a,\bar{b})$ on $\mathcal{C}_0$ defined by $\{Q(a,\bar{b}) \mid (a,b) \in \mathcal{C}_0\}$ and designate by $moy_0(Q)$ and $var_0(Q)$ the mean and the variance of this empirical distribution. Under these conditions, the globally reduced index can be written as follows

$$Q^{g0}(a,\bar{b}) = \frac{Q(a,\bar{b}) - moy_0(Q)}{\sqrt{var_0(Q)}} \tag{8}$$

Its empirical distribution is standardized ($mean = 0$ and $variance = 1$). Consequently, the following distribution of the probabilistic indices

$$\{\mathcal{I}^0(a \rightarrow b) = \Phi(-Q^{g0}(a,\bar{b})) \mid (a,b) \in \mathcal{C}_0\} \tag{9}$$

becomes finely discriminant in order to mutually compare the involved association rules. $\mathcal{I}^0(a \rightarrow b)$ defines the *Contextual Likelihood of the Link Implication* $a \rightarrow b$, with respect to $\mathcal{C}_0$.

3.2 *Three Versions of* TVe

3.2.1 The TVe Approach Based on the Mean of p-Values $VTeBarImpP$

The idea proposed in [23, 26] consists in substituting the initial n sample defined by the object set $\mathcal{O}$, a reduced and synthetic virtual sample sized by $e = 100$. More concretly, the proposed solution consists firstly, in replacing the contingency table Table 2, where n is assumed "large" by that of Table 3, respecting the proportions $p(a \wedge b)$, $p(a \wedge \bar{b})$, $p(\bar{a} \wedge b)$ and $p(\bar{a} \wedge \bar{b})$ (see Section 2) and where the total number of observations is reduced to 100.

However, the entries of Table 3 are not necessarily integer numbers they are generally rational numbers expressed by decimal approximations. The $(a \wedge \bar{b})$ entry and

Table 2 Contingency table 2 × 2

	a	$\bar{a}$	Total
b	$n(a \wedge b)$	$n(\bar{a} \wedge b)$	$n(b)$
$\bar{b}$	$n(a \wedge \bar{b})$	$n(\bar{a} \wedge \bar{b})$	$n(\bar{b})$
Total	$n(a)$	$n(\bar{a})$	n

Table 3 Contingency table 2×2 reduced to 100

	a	$\bar{a}$	Total
b	$100 \times p(a \wedge b)$	$100 \times p(\bar{a} \wedge b)$	$100 \times p(b)$
$\bar{b}$	$100 \times p(a \wedge \bar{b})$	$100 \times p(\bar{a} \wedge \bar{b})$	$100 \times p(\bar{b})$
Total	$100 \times p(a)$	$100 \times p(\bar{a})$	100

the corresponding marginal entries are reconsidered. We define three new parameters γ, α and $\bar{\beta}$ as follows: $\gamma = 100 \times p(a \wedge \bar{b})$, $\alpha = 100 \times p(a)$, and $\bar{\beta} = 100 \times p(\bar{b})$, respectively. The 8 nearest vectors with positive integer components relative to $(\gamma, \alpha, \bar{\beta})$ are considered. Each of them induces a contingency table with integer entries and for which the total number is adjusted to 100. Let us give now the technique which enables us to obtain these vectors. For this, consider the following 8 vectors of the boolean space $\{0,1\}^3$: $(0,0,0)$, $(0,0,1)$, $(0,1,0)$, $(0,1,1)$, $(1,0,0)$, $(1,0,1)$, $(1,1,0)$ and $(1,1,1)$. With each of them a contingency table with integer entries is associated. More precisely, by denoting $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$ a generic vector of $\{0,1\}^3$, the vector associated with $(\gamma, \alpha, \bar{\beta})$ is defined as follows: If ε_j is equal to 0 (resp., 1), then the j^{th} component of $(\gamma, \alpha, \bar{\beta})$ is replaced by its integer part (resp., by its integer part plus 1), $1 \leq j \leq 3$. In these conditions and even if γ, α or $\bar{\beta}$ are integers, there are always 8 neares vectors with integer components in the neighborhood of $(\gamma, \alpha, \bar{\beta})$.

The notion of a $p-values$ associated with a given entry of a 2×2 classical contingency table with integer entries is well established. This case is referred when we associate with a contingency table having positive decimal entries, the 8 nearest classical contingency tables with integer entries. With each of the latter, the $p-value$ of the $(a, \bar{b})$ cell is calculated. Thus 8 $p-values$ are obtained. A weighted mean of these define a global $p-value$. The weighting proposed is called *barycentric* mean. It is that for which the given vector $(\gamma, \alpha, \bar{\beta})$ is retrieved from its 8 nearest vectors (see above). The global $p-value$ concerned is designated by $VT100ImpBarP$ (“ValeurTest100 Implicative par moyenne Barycentrique et pour le modèle de Poisson”).

For more clarity, let us now illustrate this notion of *barycentric* mean with an example. Consider the following 2×2 contingency table Table 4 for which $n = 273$, $n(a \wedge \bar{b}) = 28$, $n(a) = 76$ and $n(\bar{b}) = 153$.

Table 4 Observed contingency table 2×2

	a	$\bar{a}$	Total
b	48	72	120
$\bar{b}$	28	125	153
Total	76	197	273

Table 5 Reduced contingency table 2×2

	a	$\bar{a}$	Total
b	17.58	26.38	43.96
$\bar{b}$	<u>10.25</u>	45.79	<u>56.04</u>
Total	<u>27.83</u>	72.17	100

The table where the total number of observations is reduced to 100 is given in Table 5.

For this $(\gamma, \alpha, \bar{\beta}) = (10.25, 27.83, 56.04)$. Thus, the nearest 8 vectors of $(\gamma, \alpha, \bar{\beta})$ with integer components are:

$$(10,27,56), (10,27,57), (10,28,56), (10,28,57),$$
$$(11,27,56), (11,27,57), (11,28,56), (11,28,57).$$

Consider now the following numbers each comprised between 0 and 1: $x = 56.04 - 56 = 0.04$, $y = 27.83 - 27 = 0.83$ and $z = 10.25 - 10 = 0.25$. Clearly, $[x + (1-x)] \times [y + (1-y)] \times [z + (1-z)]$ is equal to unity. Its expansion gives rise to the following 8 numbers comprised each between 0 and 1:

$$\left\{[(1-\varepsilon) + (-1)^{1+\varepsilon} \times z] \times [(1-\eta) + (-1)^{1+\eta} \times x] \times [(1-\zeta) + (-1)^{1+\zeta} \times y] \right.$$
$$\left. |(\varepsilon, \eta, \zeta) \in \{0,1\}^3 \right\} \qquad (10)$$

These numbers will appear below as multiplicative coefficients.

The nearest 8 vectors of $(\gamma, \alpha, \bar{\beta})$ with integer components can be expressed as follows:

$$\left\{([\gamma + \varepsilon], [\alpha + \eta], [\bar{\beta} + \zeta]) | (\varepsilon, \eta, \zeta) \in \{0,1\}^3 \right\}$$

where, for a positive real r, $[r]$ indicates the integer part of r.

As expressed above, the ordered sequence [see just above] of 8 vectors with integer components is obtained from the following $(\varepsilon, \eta, \zeta)$ 8 vectors: (0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0) and (1,1,1).

Thus, for $((\varepsilon, \eta, \zeta)) = (0,0,0)$, we obtain $(10,27,56)$, similarly, for $((\varepsilon, \eta, \zeta)) = (0,0,1)$, we obtain $(10,27,57)$, ..., and for $((\varepsilon, \eta, \zeta)) = (1,1,1)$, we obtain $(11,28,57)$.

Now, consider the 8 coefficients [see Equation (10)] determined above from x, y and z. They define the weighted *barycentric* mean of the 8 vectors we have just expressed. The obtained mean vector is the initial vector $(\gamma, \alpha, \bar{\beta})$. By illustrating this property with our example we obtain:

$$0.75 \times 0.17 \times 0.96 \times (10,27,56) + 0.75 \times 0.17 \times 0.04 \times (10,27,57) +$$
$$0.75 \times 0.83 \times 0.96 \times (10,28,56) + 0.75 \times 0.83 \times 0.04 \times (10,28,57) +$$
$$0.25 \times 0.17 \times 0.96 \times (11,27,56) + 0.25 \times 0.17 \times 0.04 \times (11,27,57) +$$
$$0.25 \times 0.83 \times 0.96 \times (11,28,56) + 0.25 \times 0.83 \times 0.04 \times (11,28,57) =$$
$$(10.25, 27.83, 56.04)$$

From Proposition 1 the $p-value$ associated with the cell $(a,\bar{b})$ of a given contingency table is obtained by $p = \Phi[Q(a,\bar{b})]$. In these conditions the 8 $p-values$ associated with the cell $(a,\bar{b})$ of the 8 contingency tables are

$$0.117,\ 0.106,\ 0.0954,\ 0.0858,\ 0.1759,\ 0.1606,\ 0.1455 \text{ and } 0.1316,$$

respectively. $VT100ImpBarP$ is the weighted *barycentric* mean of these values. We obtain 0.1115.

3.2.2 The TVe Approach Based on a Set Theoretic Correlation $VTeImpCorP$

Let us denote the set $\mathcal{O}$ of objets by $\{o_i \mid 1 \leq i \leq n\}$ and introduce the indicator functions of the subsets $\mathcal{O}(a)$ and $\mathcal{O}(b)$ (see Section 2) that we denote by a and b without any risk of ambiguity: $a(i) = 1$ (resp. 0) if and only if the boolean attribute a is $TRUE$ (resp. *FALSE*) on the object o_i, $1 \leq i \leq n$. Accordingly, $b(i) = 1$ (resp. 0) if and only if the boolean attribute b is $TRUE$ (resp. *FALSE*) on the object o_i, $1 \leq i \leq n$. 1 and 0 are considered here as numerical values. Under these conditions, the "raw" index $n(a \wedge \bar{b})$ can be written as

$$n(a \wedge \bar{b}) = \sum_{i=1}^{n} a(i) \times [1 - b(i)] \tag{11}$$

The associated random index is shown to be expressed by [17]

$$n(a^\star \wedge \bar{b^\star}) = \sum_{i=1}^{i=n} a[\sigma(i)] \times (1 - b[\tau(i)]) \tag{12}$$

For the simplest version of the random model σ and τ are two independent random permutations taken from the set G_n of all permutations on $\{1,\ldots,i,\ldots,n\}$ ($card(G_n) = n!$), provided by a uniform probability measure. In order to clarify the meaning of the random variable $n(a^\star \wedge \bar{b^\star})$ let us consider the case where $n = 8$. In this, $card(G_8) = 8! = 40320$. Two possible instances of σ and τ could be $(4,3,1,8,6,7,2,5)$ and $(7,4,5,2,8,3,6,1)$, respectively. In this notation $\sigma(1) = 4, \sigma(2) = 3, \ldots, \sigma(8) = 5$ (resp., $\tau(1) = 7, \tau(2) = 4, \ldots, \tau(8) = 1$). The associated value of $n(a^\star \wedge \bar{b^\star})$ with these permutations is $a_4 \times b_7 + a_3 \times b_4 + a_1 \times b_5 + a_8 \times b_2 + a_6 \times b_8 + a_7 \times b_3 + a_2 \times b_6 + a_5 \times b_1$.

The probability distribution of $n(a^\star \wedge \bar{b}^\star)$ under this permutational random model is determined mathematically [17]. It is a hypergeometric distribution. An adaptation of this random model has to be built in order to obtain a Poisson distribution for the probability law of $n(a^\star \wedge \bar{b}^\star)$ [10, 17].

To define $VT_eImpCorP$ where $e = 100$, the starting point is the "raw" index $100 \times \big(n(a \wedge \bar{b})/n = 100 \times p(a \wedge \bar{b})\big)$ defined by the entry $(a,\bar{b})$ in the above Table 3. Then, a set Ω of 100 objects is assumed: $\Omega = \{\omega_j \mid 1 \leq j \leq 100\}$. Without explicitly defining these objects, we define two numerical attributes $\mathcal{A}$ and $\mathcal{B}$ taking their values in the interval $[0,1]$ and such that

$$\begin{aligned}\sum_{1\leq j\leq 100} \mathcal{A}(j)\times(1-\mathcal{B}(j)) &= 100\times\big(n(a\wedge\bar{b})/n\big)\\ \sum_{1\leq j\leq 100} \mathcal{A}(j) &= 100\times\big(n(a)/n\big)\\ \sum_{1\leq j\leq 100} \mathcal{B}(j) &= 100\times\big(n(b)/n\big)\end{aligned} \tag{13}$$

The means of the numerical attributes $\mathcal{A}$ and $\mathcal{B}$ are equal to $n(a)/n$ and $n(b)/n$, respectively. This construction can be performed in different ways. The appropriate one maximizes both variances of $\mathcal{A}$ and $\mathcal{B}$. Indeed, in this way we obtain $var(\mathcal{A}) = var(a)$ and $var(\mathcal{B}) = var(b)$. Thus we establish (see [22]) that $Q(\mathcal{A},\bar{\mathcal{B}}) = \sqrt{\frac{99}{n-1}}\times Q(a,\bar{b})$. Consequently, the probabilistic index of the *Likelihood of the Implicative Link* becomes

$$\mathcal{I}(a\rightarrow b) = Pr\{n(\mathcal{A}^{\star}\wedge\bar{\mathcal{B}}^{\star}) > n(\mathcal{A}\wedge\bar{\mathcal{B}})\} = \Phi(-Q(\mathcal{A},\bar{\mathcal{B}})) \tag{14}$$

3.2.3 The *TVe* Approach Based on a Reduction by Projection on a Random Set of Size *e VTeImpProj*

In fact, as given in [23, 26], the intuitive presentation of *TVe* ($e = 100$) is conceptually completely disconnected from *VTeImpBarP* (see Section 3.2.1) considered to approximate it. Indeed, in the mentioned presentation it is proposed to compute the average of the index $-Q(a,\bar{b})$ on a sequence $(E^{(1)},E^{(2)},\ldots,E^{(l)},\ldots,E^{(L)})$ of L independent random samples, each with size e. L is supposed to be large enough. Besides, the considered sampling of a given $E^{(l)}$ $1\leq l\leq L$ is made without replacement. As admitted in [23] this process is not straightforward to implement and requires in addition the determination of a relevant value for L. In fact a mathematical and statistical computation can be substituted for it, providing an efficient and simple solution. The mathematical expression of the proposed index is

$$\frac{1}{L}\sum_{1\leq l\leq L}\frac{card[\mathcal{O}(a)\cap\mathcal{O}(\bar{b})\cap E^{(l)}] - \frac{card[\mathcal{O}(a)\cap E^{(l)}]\times card[\mathcal{O}(\bar{b})\cap E^{(l)}]}{card(E^{(l)})}}{\sqrt{\frac{card[\mathcal{O}(a)\cap E^{(l)}]\times card[\mathcal{O}(\bar{b})\cap E^{(l)}]}{card(E^{(l)})}}} \tag{15}$$

The mathematical expression which has to be substituted for it is

$$\mathcal{E}\Big(\frac{card[\mathcal{O}(a)\cap\mathcal{O}(\bar{b})\cap E^{\star}] - \frac{card[\mathcal{O}(a)\cap E^{\star}]\times card[\mathcal{O}(\bar{b})\cap E^{\star}]}{e}}{\sqrt{\frac{card[\mathcal{O}(a)\cap E^{\star}]\times card[\mathcal{O}(\bar{b})\cap E^{\star}]}{e}}}\Big) \tag{16}$$

where $\mathcal{E}$ designates the mathematical expectation and where $E^\star$ defines an $\mathcal{O}$ random subset of cardinality e taken in the set of all $\mathcal{O}$ subsets with the same cardinal e, endowed with a uniform probability measure, [22]. For analytical complexity reasons the adopted solution is

$$\frac{\mathcal{E}\left(card[\mathcal{O}(a)\cap\mathcal{O}(\bar{b})\cap E^\star] - \frac{card[\mathcal{O}(a)\cap E^\star]\times card[\mathcal{O}(\bar{b})\cap E^\star]}{e}\right)}{\sqrt{\mathcal{E}\left(\frac{card[\mathcal{O}(a)\cap E^\star]\times card[\mathcal{O}(\bar{b})\cap E^\star]}{e}\right)}} \tag{17}$$

In this equation, with respect to (16), we substitute for the random indices $card[\mathcal{O}(a)\cap E^\star]$ and $card[\mathcal{O}(\bar{b})\cap E^\star]$, their respective mathematical expectations.

The following expression is proved in [22] for the above expression (17)

$$\frac{\frac{1}{\sqrt{n(n-1)}}\times[(ne-1)\times n(a\wedge\bar{b})-(e-1)\times n(a)\times n(\bar{b})]}{\sqrt{(n-e)\times n(a\wedge\bar{b})+(e-1)\times n(a)\times n(\bar{b})}} \tag{18}$$

The *Test Value* is defined by the value of this expression multiplied by -1.

4 Variations of $VLgrImpP$ and $VTeImpCorP$ for Increasing the Size of the Object Set

4.1 The Variational Models M_1 and M_2

The M_1 model is a classical one. For a given boolean attribute pair (a,b), the cardinalities n , $n(a)$, $n(\bar{a})$, $n(b)$, $n(\bar{b})$, $n(a\wedge b)$, $n(a\wedge\bar{b})$, $n(\bar{a}\wedge b)$ and $n(\bar{a}\wedge\bar{b})$, are multiplied by the same integer value k, where k increases from its initial value 1. The previous cardinalities become

$$\begin{gathered} k\times n\,,\, k\times n(a)\,,\, k\times n(\bar{a})\,,\, k\times n(b)\,,\, k\times n(\bar{b})\,, \\ k\times n(a\wedge b)\,,\, k\times n(a\wedge\bar{b})\,,\, k\times n(\bar{a}\wedge b) \text{ and } k\times n(\bar{a}\wedge\bar{b}) \end{gathered} \tag{19}$$

This increasing model which preserves the respective proportions $p(a)$,$p(\bar{a})$,$p(b)$, $p(\bar{b})$,$p(a\wedge b)$,$p(a\wedge\bar{b})$,$p(\bar{a}\wedge b)$ and $p(\bar{a}\wedge\bar{b})$ (see Section 2), was considered in [23, 26].

For the second model M_2 which is a specific one, the cardinalities $n(a\wedge b)$, $n(a)$ and $n(b)$ are constant, only n increases from its initial value defined with respect to a real case. This technique can be compared to adding occurences for which both attributes a and b have a *FALSE* value. By denoting x the positive integer corresponding to the increasing in n, the previous cardinalities become

$$\begin{gathered} n+x\,,\, n(a)\,,\, n(\bar{a})+x\,,\, n(b)\,,\, n(\bar{b})+x\,, \\ n(a\wedge b)\,,\, n(a\wedge\bar{b})\,,\, n(\bar{a}\wedge b) \text{ and } n(\bar{a}\wedge\bar{b})+x \end{gathered} \tag{20}$$

The relevancy of this model is justified by the fact that generally, the number of elements for which a given boolean attribute is $TRUE$ is very small in relation to the database size, which is usually very large. Therefore, the variations in the different indices will be compared under the increasing model M_2. Two facets will be considered for this comparison: *theoretical* and *experimental*. The behaviour of the two indices $VLgrImpP$ and $VTeImpCorP$ will be studied analytically. The latter index is taken as a prototype for the TVe indices. It is among the three TVe indices the most appropriate for an analytical study. The experimental analysis will be considered in Section 5. A global view of the respective behaviour of the four indices ($VLgrImpP$ and the three versions of TVe), will be given in that section.

4.2 Behaviour of $VLgrImpP$ and $VTeImpCorP$ under the Models M_1 and M_2

4.2.1 Behaviour with Respect to the Increasing Model M_1

By denoting ν the total number of observations of the contingency table - crossing $\{a,\bar{a}\}$ with $\{b,\bar{b}\}$ - obtained by applying an instance of the model M_1 (resp., M_2), the new contigency table, reduced to 100 observations, is obtained by multiplying all the entries by $100/\nu$. A given instance of the index $Q(a,\bar{b})$ (see (3)) for the model M_1 (resp., M_2) is computed on the basis of this new contingency table.

Proposition 2. *The index $VLgrImpP$ is invariant for the increasing model M_1.*

Proposition 3. *The index $VTeImpCorP$ is invariant for the increasing model M_1.*

These results are straightforward to prove.

4.2.2 Behaviour of $VLgrImpP$ with Respect to the Increasing Model M_2

Equation (20) defines the variations of the integer numbers in the cells of the contingency table crossing $\{a,\bar{a}\}$ and $\{b,\bar{b}\}$. Let us denote by $Q_x(a,\bar{b})$ the corresponding index (see equation (3)) $Q_x(a,\bar{b})$ which can be written as follows

$$Q_x(a,\bar{b}) = \frac{n(a \wedge \bar{b}) - \frac{n(a)\times[n(\bar{b})+x]}{(n+x)}}{\{\frac{n(a)\times[n(\bar{b})+x]}{(n+x)}\}^{\frac{1}{2}}} \qquad (21)$$

It gives rise to an increasing function of the *Likelihood of the Link Implication* probabilistic index. More precisely, the behaviour of $-Q_x(a,\bar{b})$ with respect to the variation of x, leads us to establish the following result.

Proposition 4. *$-Q_x(a,\bar{b})$ is an increasing function with respect to x, its variation rate decreases with respect to x.*

In order to clarify the proof of this statement given in [22] (page 38), we simplify our notations by setting $\gamma = n(a \wedge \bar{b})$, $\alpha = n(a)$, $\beta = n(b)$, $\bar{\beta} = n(\bar{b})$ and $y = n(\bar{b}) + x$. On the other hand, if we denote $-Q_x(a,\bar{b})$ by $\Phi(y)$,we obtain from Equation (21)

$$\Phi(y) = \frac{-\gamma + \frac{\alpha \times y}{\beta + y}}{\sqrt{\frac{\alpha \times y}{\beta + y}}} \tag{22}$$

It is easy to show that the first and second derivatives of this function $\Phi'(y)$ and $\Phi''(y)$ are, respectively, positive and negative functions for all values of y ($\Phi'(y) > 0$ and $\Phi''(y) < 0$) (for more details see the above reference). Therefore, $\Phi(y)$ and $\Phi'(y)$ are an increasing and a decreasing functions, respectively. Consequently the property expressed in Proposition 4, follows.

Let us give here a sequence of values of $-Q_x(a,\bar{b})$ obtained in the case of an example provided by the database "Adult" available on the site "UCI Machine Learning Repository" which were used in [23, Section 5]. For this, $n = 14,743$, $n(a) = 4,819$, $n(\bar{b}) = 3,522$ and $n(a \wedge \bar{b}) = 225$.

Table 6 Increasing behaviour of $-Q_x(a,\bar{b})$

x	$-Q_x(a,\bar{b})$	rate of $-Q_x(a,\bar{b})$
0	27.712	
1000	31.599	3.887
2000	34.687	3.088
10000	47.503	1.602
50000	60.234	0.318
100000	63.233	0.060

Consider now the normalized version $Q^{g0}(a,\bar{b})$ (8) of $-Q_x(a,\bar{b})$ (21) and the associated "*Contextual Likelihood of the Link Implication*" $\Phi[Q^{g0}(a,\bar{b})]$. The increasing of $-Q_x(a,\bar{b})$ does not entail the same property for its normalized version. However and in practice, the variation of $\Phi[Q^{g0}(a,\bar{b})]$ appears as globally monotonic and that, in a consistent manner depending on the initial comparison configuration ($x = 0$) (see Section 5).

Behaviour of $VT100ImpCorP$ with respect to the increasing model M_2

Let us indicate by $-Q_x^{100}$ the adopted version of $VT100$ index. We obtain

$$-Q_x^{100}(a,\bar{b}) = -10 \times \frac{[n(a \wedge \bar{b}) - \frac{n(a) \times [n(\bar{b}) + x]}{n+x}]}{\sqrt{n(a) \times [n(\bar{b}) + x]}} \tag{23}$$

Now, with the above introduced notations - to a multiplicative coefficient 10 - by denoting $\Psi(y) = -Q_x^{100}(a,\bar{b})$, we have

$$\Psi(y) = \frac{-\gamma + \alpha \times \frac{y}{\beta+y}}{\sqrt{\alpha \times y}} \tag{24}$$

The variation analysis of this index, interpreted as a function of y, leads to the study of the following second trinomial in y: $(\gamma - \alpha) \times y^2 + \beta \times (\alpha + 2\gamma) \times y + \gamma \times \beta^2$. Indeed, the derivative $\Psi'(y)$ of $\Psi(y)$ is obtained by multiplying the latter expression with a positive function of y (the exact calculation of $\Psi'(y)$ is left for the reader). And then, only the sign of that quadratic polynom is of concern. The associated discriminant can be put in the following form: $\Delta = \beta^2 \times [(\alpha + 2\gamma)^2 + 4(\alpha - \gamma)]$. Since $\gamma = n(a \wedge \bar{b}) < \alpha = n(a)$, it is strictly positive. The two roots can be written

$$y' = \frac{\beta(\alpha + 2\gamma) + \sqrt{\Delta}}{2(\alpha - \gamma)}$$
$$y'' = \frac{\beta(\alpha + 2\gamma) - \sqrt{\Delta}}{2(\alpha - \gamma)}$$

Trivially $y' > 0$ and $y'' \leq 0$. Because $\gamma < \alpha$ the trinomial is positive for $y \leq y'$ and negative for $y > y'$. Therefore, $\Psi(y)$ is increasing (resp., decreasing) for $0 \leq x \leq y' - n(\bar{b})$ (resp., $x > y' - n(\bar{b})$). Consequently, we obtain the following result

Proposition 5. $-Q_x^{100}(a,\bar{b})$ *is increasing for x varying in the interval* $[0, y' - n(\bar{b})]$ *and decreasing for x greater than* $y' - n(b)$.

To illustrate this, let us go back to the above above example of the database "Adult". One obtains the results in Table 7.

Such a behaviour of $-Q_x^{100}(a,\bar{b})$, increasing first and decreasing next, may seem surprising. In fact, it has been observed in the experimental analysis (see Section 5).

Table 7 Variation of $-Q_x^{100}(a,\bar{b})$ with respect to x

x	$-Q_x^{100}(a,\bar{b})$
0	2.282
1000	2.518
2000	2.681
10000	3.020
15000	2.973
20000	2.887
30000	2.694
50000	2.367
100000	1.982

5 Experimental Analysis

For a given rule $a \rightarrow b$ associated with an ordered pair (a,b) of boolean attributes, we are going to consider different statistical dependency configurations. Each of them is defined from the respective values of $n(a \wedge b)$, $n(a \wedge \bar{b})$, $n(\bar{a} \wedge b)$ and $n(\bar{a} \wedge \bar{b})$ (see the beginning of Section 2). The dependency configurations could have been built mathematically without any relation with a real database. We have chosen to obtain these configurations in the context of two well known databases *Wages* [3] and *Abalone* [2]. The Wages[1] database consists of 534 objects described by 11 attributes including 4 quantitative attributes: Education (*number of years of education*), Experience (*number of years of work experience*), Wage (*dollars per hour*) and Age (*years*). Categorical attributes are: Region (*person who lives in the South or elsewhere*), Sex, Union membership, Race (*Hispanic, White and Other*), Occupation (*Management, Sales, Clerical, Service, Professional and Other*), Sector (*Manufacturing, Construction and Other*) and Married (*marital status*). Each of the quantitative (numerical) attributes is discretized leading to a categorical attribute. Discretizing numerical attributes is a very important problem for which more or less sophisticated methods have been developed [19, 25]. Taking into account the nature of our data, the principle of obtaining categories as balanced as possible according to their respective sizes, was adopted. Thus, a set of boolean attributes associated with the obtained categories is determined. This scale conversion after categorization of numerical attributes is often called "complete disjunctive coding". We obtain for the Wages database 40 boolean attributes. The Abalone[2] database consists of 4,177 objects described by 9 attributes including 8 quantitative attributes: Length, Diameter, Height, Whole weight, Shucked weight, Viscera weight, Shell weight and Rings. Categorical attribute is Sex (*Male, Female and Sex=Infant*). After categorization of the numerical attributes and the associated complete disjunctive coding, the obtained database includes 43 boolean attributes.

Four fundamental configurations will be considered. They can be distinguished by means of the index $d(a,b) = p(a \wedge b)/(p(a) \times p(b))$ (see Section 2 and 3.2.1 for notations). Conceptually this index is a density of probability. It was called *Lift* index in the "Data Mining" domain [9]. For (a,b) the four configurations are defined by

- Incompatibility between a and b ($d(a,b) = 0$)
- Repulsion between a and b ($d(a,b) < 1$)
- Independence between a and b ($d(a,b) = 1$)
- Attraction between a and b ($d(a,b) > 1$)

5.1 Incompatibility

The first studied rules are "$\textit{education} = [17,18] \rightarrow \textit{worker}$" (Wages database) and "$\textit{length} =]0.223;0.371] \rightarrow \textit{diameter} =]0.412;0.531]$" (Abalone database). These

[1] This database can be uploaded to "http://www.isima.fr/~guillaum/".

[2] This database is available on the site "UCI Machine Learning Repository" (http://kdd.ics.uci.edu/).

rules illustrate the case of incompatibility. Indeed, the contingency of the attributes $a =$ “$education = [17,18]$” and $b =$ “$worker$” for the Wages database is given by $n(a \wedge b) = 0$, $n(a) = 55$ and $n(b) = 83$ and the contingency of the attributes $a =$ “$length =]0.223;0.371]$” and $b =$ “$diameter =]0.412;0.531]$” for the Abalone database is given by $n(a \wedge b) = 0$, $n(a) = 451$ and $n(b) = 1,951$.

The variation of the four measures $VLgrImpP$, $VTeImpBarP$, $VTeImpCorP$ and $VTeImpProj$ according to the increasing model M_1 are given in Figure 1. The nature of this variation is clearly different when the increasing model is M_2, see Figure 2.

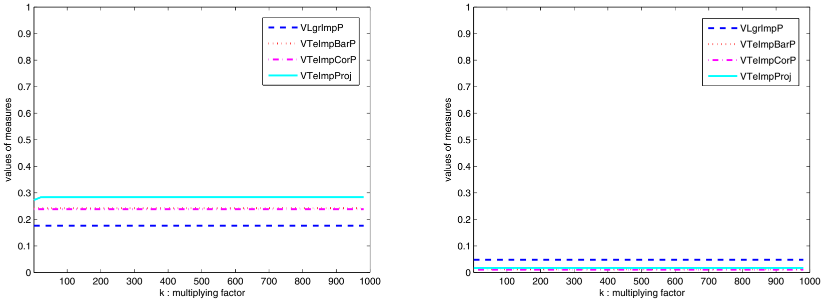


Fig. 1 Comparison of measure variations according to the model M_1 for rules “$education = [17,18] \rightarrow worker$” (left part of figure) and “$length =]0.223;0.371] \rightarrow diameter =]0.412;0.531]$” (right part of figure)

First, an invariance of the different measures can be verified when all numbers are multiplied by a coefficient k (see Figure 1) (see also Propositions 2 and 3). However, for very small values of k, there is a slight growth in the $VTeImpProj$ curve (see left part of Figure 1). We will find a similar growth on all the following curves, and for all studied cases. We observe a similar behaviour for the measures $VTeImpBarP$ and $VTeImpCorP$ since their curves are practically overlapped. Now, let us compare the Incompatibility case (see Figure 1) with the Independence case (see Figure 5) and the Attraction one (see Figure 7). All of the four measures are shown to be selective for both databases. This means that their respective values are very low and the gap for each of them between the Incompatibility case (Figure 1) and the independence case (Figure 5) (resp., Attraction case (Figure 7)) is large. For the Wages database and with respect to the Independence case, the most selective measure is $VLgrImpP$. For the Abalone database there are two equivalently most selective measures: $VTeImpProj$ and $VLgrImpP$. The size of the Abalone database and the contingency of the rule “$length =]0.223;0.371] \rightarrow diameter =]0.412;0.531]$” lead to very low values for the four measures (see right part of Figure 1). Notice that when the size of the database increases (see right part of Figure 2), the measure $VLgrImpP$ remains selective.

Now, a comparison with the Attraction case leads to analogous results for the Wages database. For the latter, clearly, $VLgrImpP$ is the most selective. For the Abalone database, pratically all of the four measures are equally very selective.

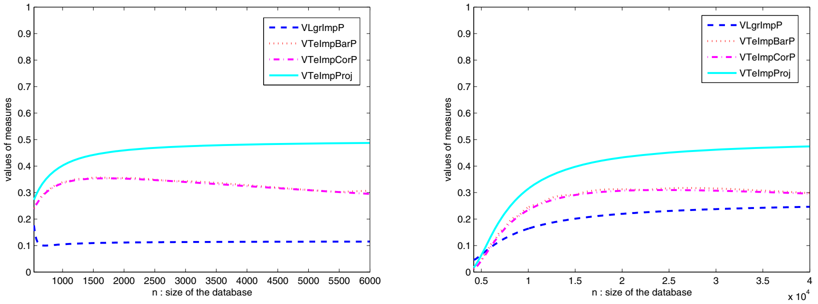


Fig. 2 Comparison of measures variations according to the model M_2 for rules "$education = [17,18] \to worker$" (left part of figure) and "$length =]0.223;0.371] \to diameter =]0.412;0.531]$" (right part of figure)

In the case of incompatibility and for the model M_2 (see Figure 2), we observe a similar behaviour for the measures $VTeImpBarP$ and $VTeImpCorP$ since their curves are practically overlapping as in the case of the model M_1 (see Figure 1). These curves first increase and then decrease as has been indicated in Proposition 5. As far as the measure $VTeImpProj$ is concerned, it increases to tend towards the value 0.5 for the two rules. For this Incompatibility case (under the model M_2), the global variation tendency of the measure $VTeImpProj$ is continually increasing. This behaviour is different from the other studied configurations where $VTeImpProj$ is first increasing and then decreasing (see Figures 4, 6, 8). For this Incompatibility case, the measure $VLgrImpP$ has a similar behaviour to that of $VTeImpProj$, that is to say and except at the very beginning, a continually and slowly increasing variation tendency. And this behaviour can be noticed for all of configurations (see Figures 4, 6, 8). Now, we can observe the tendency of the value of $VLgrImpP$ towards 0.1 for the Wages database and 0.2 for the Abalone database. This difference is mainly due to the cardinalities defining the entries of the contingency tables in both cases. In order to realize this point let us go back to the coefficient $Q_x(a,\bar{b})$ (see Equation (21)). By considering for the Wages (resp., Abalone) database $x = 0, 1466$, and $3,966$ (resp., $0, 11823$ and $31,823$) we obtain for $-Q_x(a,\bar{b}) : 6.82, 7.26$ and 7.35 (resp., $15.50, 19.70$ and 20.65). Thus, the global normalization effect (see (8)) is similar for both data bases.

5.2 Repulsion

The rules that we are going to study are "$union\ member \to sex = female$" (Wages database) and "$Sex = infant \to diameter =]0.412;0.531]$" (Abalone database). The contingency elements of this first rule are: $n(a \wedge b) = 28$, $n(a) = 96$ and $n(b) = 245$; and for this second rule are: $n(a \wedge b) = 243$, $n(a) = 1,342$ and $n(b) = 1,951$. In these two cases, the occurence of the event "*union member*" reduces the chance of observing the event “$sex = female$”; and the occurence of the event “$sex = infant$” reduces the chance of observing the event “$diameter =]0.412;0.531]$”. Indeed, for the first rule, the observed number of persons satisfying both the antecedent “*union member*” and the consequent “$sex = female$” is 28 while the expected number under the independence hypothesis is equal to $96 \times 245/534 = 44$. For the second one, the observed number of persons statisfying both the antecedent “$Sex = infant$” and the consequent “$diameter =]0.412;0.531]$” is 243 while the expected number under the independence hypothesis is equal to $1,342 \times 1,951/4,177 = 627$.

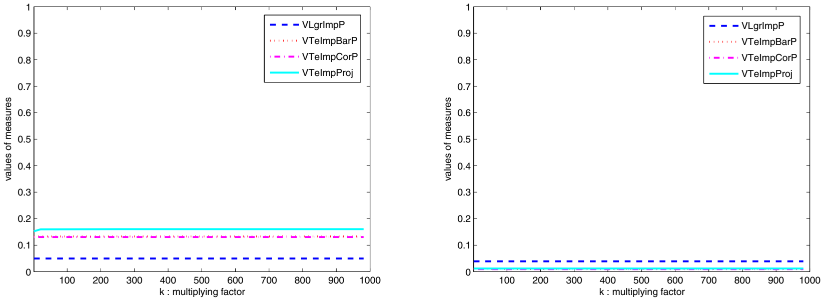


Fig. 3 Comparison of measures variations according to the model M_1 for rules “$union\ member \rightarrow sex = female$” (left part of figure) and “$Sex = infant \rightarrow diameter =]0.412;0.531]$” (right part of figure)

We can verify the insensivity of measures when the coefficient k increases, according to the model M_1. Moreover, we can notice in Figure 3 as in Figure 1 associated with the incompatibility case (“$education = [17,18] \rightarrow worker$”) a similar behaviour for the four measures: the curves of the measures $VTeImpBarP$ and $VTeImpCorP$ are practically overlapping, the lowest value is observed for the measure $VLgrImpP$ and the highest one for $VTeImpProj$. The same general behaviour is observed for the Abalone database, under the M_1 model. However, for the latter, the value of the measure $VLgrImpP$ is the highest one. In any case, all of the four index values are very low and this can be understood by the relative large size of the Abalone database with respect to that of the Wages database.

It is perhaps surprising to observe that for this configuration of statistical repulsion between “*union member*” and “$sex = female$” where the conjunction between the two concerned boolean attributes is not empty, the values of the fourth measures (see Figure 3) are lower than in the previous incompatibility case (“$education = [17,18] \rightarrow worker$”) 1. However, the incompatibility case is relative here to boolean attributes whose frequencies are very low. In the latter case the product of these frequencies is equal to $55 \times 83 = 4,565$ while in the repulsive case (“$union\ member \rightarrow sex = female$”) this product is equal to $96 \times 245 = 23,520$. If we refer to the values of the index $-Q(a,\bar{b})$, we obtain the values -1.25 for the incompatibility case and -2.23 for the repulsive case, respectively. These statistical implications are actually contrary to nature. If we considered the two complementary rules

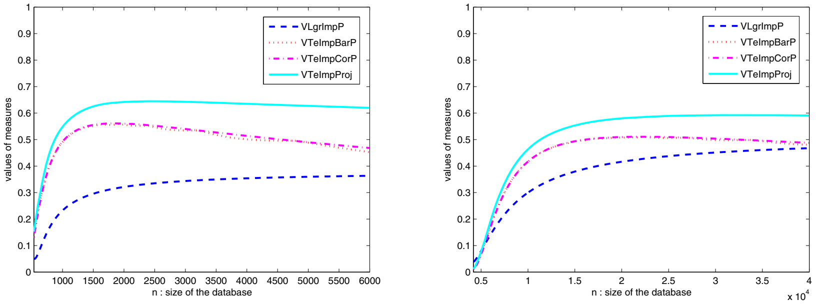


Fig. 4 Comparison of measures variations according to the model M_2 for rules "*union member* $\rightarrow$ *sex* $=$ *female*" (left part of figure) and "*sex* $=$ *infant* $\rightarrow$ *diameter* $=$ $]0.412;0.531]$" (right part of figure)

("*education* $= [17,18] \rightarrow$ *no worker*") and ("*union member* $\rightarrow$ *sex* $=$ *male*"), we obtain respectively the following values: 2.92 and 2.42.

The curves of the measures $VTeImpProj$, $VTeImpBar$ and $VTeCorP$ in Figure 4 first increase and then decrease contrary to the curve of the measure $VLgrImpP$ which does not decrease but tends towards a value close to 0.4 for the rule "*union member* $\rightarrow$ *sex* $=$ *female*" and close to 0.5 for the rule "*sex* $=$ *infant* $\rightarrow$ *diameter* $=]0.412;0.531]$". A similar behaviour for the measures $VTeImpBarP$ and $VTeImpCorP$ is observed, and the same respective selectivities for the four measures (*lowest values for* $VLgrImpP$ *and highest values for* $VTeImpProj$) can be noticed. With the exception of the first values of the measures which correspond to a low number of items in the Wages database, the values of the four measures are higher than in our previous incompatibility case. However, in this case the evolution model is very different. For this, let us go back to the coefficient $-Q_x(a,\bar{b})$ (see Equation (20)) whose overall reduction leads to $VLgrImpP(a \rightarrow b)$. Denote by $\Phi(x)$ [resp., $\Psi(x)$] this coefficient in the case of the evaluation of the rule ("*education* $= [17,18] \rightarrow$ *worker*") [resp., ("*union member* $\rightarrow$ *sex* $=$ *female*")]. We can verify that the following difference between the derivatives $\Psi'(x) - \Phi'(x)$ is positive. In this case, after the operation of overall reduction, the evolution curve associated with $VLgrImpP$ ("*union member* $\rightarrow$ *sex* $=$ *female*") (see left part of Figure 4) becomes considerably higher than that associated with ("*education* $= [17,18] \rightarrow$ *worker*").

5.3 Independence

Rules that we now study concern the case of independence. The first one is "*wage* $= [6.67;9] \rightarrow$ *north*" (Wages database) and the second one is "*viscera weight* $=]0.3043;0.4562] \rightarrow$ *rings* $=]6;12]$" (Abalone database). The contingency of this first rule is defined by $n(a \wedge b) = 71$, $n(a) = 100$ and $n(b) = 378$; and the contingency of this second one is defined by $n(a \wedge b) = 371$, $n(a) = 510$ and $n(b) = 3,036$. We still observe under the model M_1, an invariance as the values of k increase. Notice in Figure 5 that the values of the various measures are higher than in the case of incompatibility (see Figure 1) and also in the repulsion case (see Figure 3). This is reassuring because this rule is stronger than those previously considered. The measures $VTeImpBarP$ and $VTeImpCorP$ still behave in a similar way but what is new is the fact that the measure $VLgrImpP$ has the highest constant value whereas previously it had the lowest one. Nevertheless, this latter value is about 0.5, which reflects independence perfectly. The measure $VTeImpProj$ also has a value about 0.5.

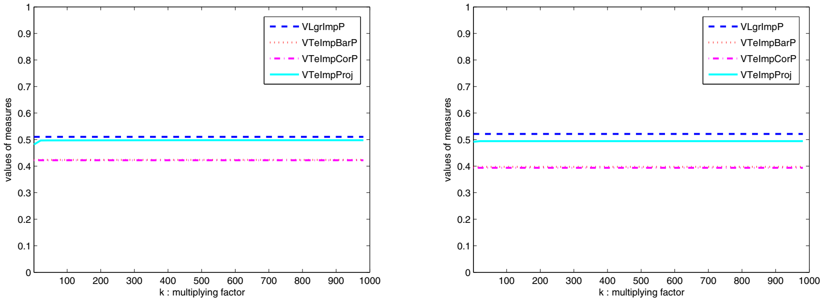


Fig. 5 Comparison of measures variations according to the model M_1 for rules "$wage = [6.67;9] \rightarrow north$" (left part of figure) and "$viscera\ weight =]0.3043;0.4562] \rightarrow rings =]6;12]$" (right part of figure)

For the evolution model M_2 see Figure 6, we observe that the curves of the measures $VTeImpProj$, $VTeImpBarP$ and $VTeImpCorP$ still increase at first and then decrease unlike the curve for the measure $VLgrImpP$ which does not decrease but tends towards a value close to 0.85. What is different compared with previous curves (see Figures 2 and 4), is that for high values of n (greater than 4000), the values of the measure $VLgrImpP$ are the highest ones. This situation is also observed for the model M_1 of this same rule.

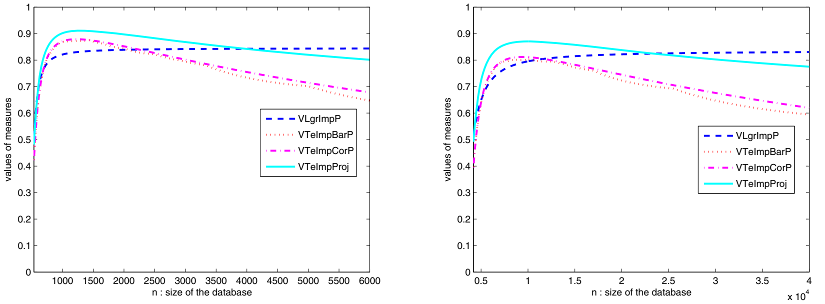


Fig. 6 Comparison of measures variations according to the model M_2 for rules "$wage = [6.67] \rightarrow north$" (left part of figure) and "$visceraweight =]0.3043;0.4562] \rightarrow rings =]6;12]$" (right part of figure)

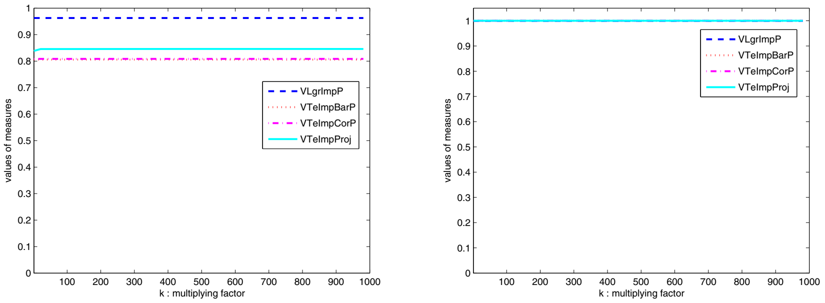


Fig. 7 Comparison of measures variations according to the model M_1 for rules "$sex = female \rightarrow non\ union\ member$" (left part of figure) and "$length =]0.371; 0.519] \rightarrow diameter =]0.293; 0.412]$" (right part of figure)

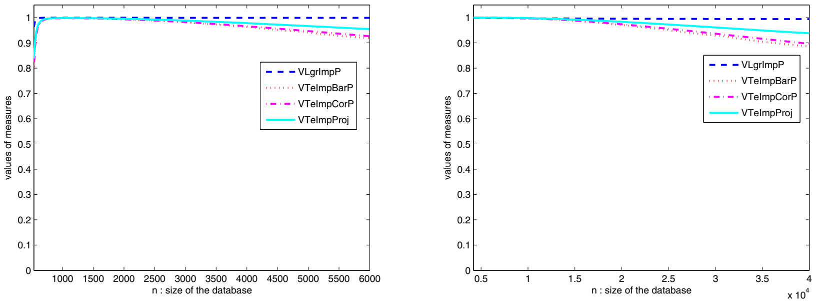


Fig. 8 Comparison of measures variations according to the model M_2 for rules "$sex = female \rightarrow non\ union\ member$" (left part of figure) and "$length =]0.371; 0.519] \rightarrow diameter =]0.293; 0.412]$" (right part of figure)

5.4 Attraction

The last studied rules are "$sex = female \rightarrow non\ union\ member$" (Wages database) and "$length =]0.371; 0.519] \rightarrow diameter =]0.293; 0.412]$" (Abalone database). These rules correspond to an attraction between the two boolean attributes a and b, because the observed number of examples ($n(a \wedge b) = 217$ for the Wages database and $n(a \wedge b) = 1,113$ for the Abalone database) is greater than the expected number in the independence hypothesis ($245 \times 438/534 = 201$ for the Wages database and $1,238 \times 1,325/4,177 = 393$ for the Abalone database). Indeed, we have $n(a) = 245$ and $n(b) = 438$ for the rule "$sex = female \rightarrow non\ union\ member$" and for the rule "$length =]0.371; 0.519] \rightarrow diameter =]0.293; 0.412]$", we have $n(a) = 1,238$ and $n(b) = 1,325$.

Figures 7 and 8 show the variation of the four measures for these two rules according to the models M_1 and M_2, respectively. Notice that we obtain similar curves as those of previous rules (*independence case*) with here the particularity that the measure $VLgrImpP$ tends very quickly towards the value 1 for the model M_2.

6 Conclusion and Prospects

As expressed above (see Section 2.1) there are two alternatives for the Likelihood Linkage index: *symmetrical* (1) and *asymmetrical* (2). For a given boolean attribute ordered pair (a,b), the symmetrical version measurement focuses on the equivalent strength between a and b whereas the asymmetrical version focuses on the implicative tendency $a \rightarrow b$. Both alternatives can be considered for capturing the intuitive notion of interestingness of an association rule. In this work we highlight the asymmetrical facet of an association rule.

The *Test Value* index appeared in the *Data Analysis* and *Data Mining* literature much more recently and without establishing the necessary connection with the *Likelihood Linkage* index. This connection is provided by Proposition 1 (see Section 2.2). Both indices are close in practice. However and conceptually, the spirit in which the Test Value index was setup is very different to that of the Likelihood Linkage index.

In any case, these types of indices referring directly to a probability scale, become non discriminant when the number of observations becomes very large (more than several hundreds). Under these conditions, a preliminary standardization is essential for discriminant probabilistic pairwise comparison between descriptive attributes. This standardization is performed according to the respective conceptual natures of the indices LL and TV. Thus, $VLgrImpP$ on one hand, and $VTeImpBarP$, $VTeImpCorP$ and $VTeImpProj$ on the other hand, have been elaborated. Then, it is important to compare their respective behaviour with respect to increasing models of the number of observations. For this purpose, the models M_1 and M_2 have been defined (see Section 4.1).

We have established theoretically and experimentally the invariance of the values of the different indices under the increasing model M_1. Besides, the observed value of $VLgrImpP$ is recognized intuitively as more consistent with the nature and the strength of the considered rule. The analysis of the variations in the different indices ($VLgrImpP$ on one hand, and $VTeImpBarP$, $VTeImpCorP$ and $VTeImpProj$ on the other hand) shows the distinctive part played by the $VLgrImpP$ index. Indeed, for the M_2 model, it is the only index to have a monotonic variation: either increasing or decreasing according to the implicative relation structure. The increasing or decreasing behaviour begins in both cases with a strong slope. The latter varies slowly and ends by tending to a horizontal slope. The experimental analysis (see Section 5 and Section 6 of [22]) has validated the theoretical study (see Section 4). This analysis shows that the $VLgrImpP$ index has much more chance of preserving a rule $a \rightarrow b$ for which $n(a)$ and $n(b)$ are weak, when the database size increases.

This study encourages us to consider newer variation models other than M_1 and M_2. In a third M_3 model, considered in [8], an additional variation parameter is considered. Only $\mathcal{O}(a)$ and $card[\mathcal{O}(b)]$ are fixed. By associating with the subset $\mathcal{O}(b)$ a subset Y of $\mathcal{O}$ – whose cardinality is $n(b)$ – and by varying Y step by step, one can go for the association between $\mathcal{O}(a)$ and Y, from incompatibility ($\mathcal{O}(a) \cap Y = \emptyset$) to the logical implication ($\mathcal{O}(a) \subset Y$). An interesting model is also proposed in [28]. In this, from the initial contingency table 2×2 crossing $\{a, \bar{a}\}$ with $\{b, \bar{b}\}$

a new table is deduced by multiplying either its two rows $[n(a \wedge b), n(a \wedge \bar{b})]$ and $[n(\bar{a} \wedge b), n(\bar{a} \wedge \bar{b})]$ or its two columns $[n(a \wedge b), n(\bar{a} \wedge b)]$ and $[n(a \wedge \bar{b}), n(\bar{a} \wedge \bar{b})]$ by two constants k_1 and k_2, respectively.

Now, let us come back to the variation model M_2. Complementary analysis allows us to go into greater depth in the comparison of the indices. First, new structural implicative configurations can be examined. Otherwise, the different indices have to be compared with respect to their abilities to discriminate between different rules, when the number of observations increases according to the model M_2. In connection with these abilities and for the selection problem of a small number of interesting rules, we have to compare the selected rules by each of the interestingness measures $VLgrImpP$, $VTeImpBarP$, $VTeImpCorP$ and $VTeImpProj$. Finally, it is of importance to study the proposed indices in comparison with measures of interestingness whose value scales are not necessarily probabilistic. All these questions are being studied.

Acknowledgements. We thank Carol Fynn for her invaluable assistance with the writing of this paper in English.

We are also very indebted to Philippe Louarn for his help in the LaTeX completion of this paper.

References

[1] Agrawal, R., Imielsky, T., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of the 8th ACM SIGKDD Conference on Knowledge Discovery and Data Mining 1993, pp. 207–216. ACM (1993)

[2] Bay, S.: The UCI KDD archive. University of California, Department of Information and Computer Science, Irvine (1999), http://kdd.ics.uci.edu/

[3] Berndt, E.: The Practice of Econometrics. Addison-Wesley, NY (1991)

[4] Daudé, F.: Analyse et justification de la notion de ressemblance entre variables qualitatives dans l'optique de la classification hiérarchique par avl. Thèse de doctorat, Université de Rennes 1 (1992)

[5] Feller, W.: An Introduction to Probability Theory and Its Applications. Wiley, New York (1968)

[6] Gras, R.: Contribution à l'étude expérimentale et à l'analyse de certaines acquisitions cognitives et de certains objectifs didactiques en mathématiques. Thèse de doctorat d'état, Université de Rennes 1 (1979)

[7] Gras, R.: L'implication statistique. La pensée sauvage, Paris (1996)

[8] Guillaume, S., Lerman, I.C.: Analyse du comportement limite d'indices probabilistes pour une sélection discriminante. In: Khenchaf, A., Poncelet, P. (eds.) EGC 2011. RNTI, vol. RNTI E. 20, pp. 657–664. Hermann (2011)

[9] IBM: Ibm intelligent miner user's guide, version 1, release 1. Tech. rep. (1996)

[10] Lagrange, J.B.: Analyse implicative d'un ensemble de variables numériques; application au traitement d'un questionnaire à réponses modales ordonnées. Revue de Statistique Appliquée 46, 71–93 (1998)

[11] Lallich, S., Teytaud, O.: Évaluation et validation de l'intérêt des règles d'association. In: Mesures de Qualité pour la Fouille des Données 2004. RNTI, vol. RNTI-E-1, pp. 193–218. Cépaduès (2004)

[12] Lenca, P., Meyer, P., Picouet, B., Lallich, S.: Évaluation et analyse multicritère des mesures de qualité des règles d'association. In: Mesures de Qualité pour la Fouille des Données 2004. RNTI, vol. RNTI-E-1, pp. 219–246. Cépaduès (2004)

[13] Lerman, I.C.: Sur l'analyse des données préalable à une classification automatique; proposition d'une nouvelle mesure de similarité. Mathématiques et Sciences Humaines 8, 5–15 (1970)

[14] Lerman, I.C.: Introduction à une méthode de classification automatique illustrée par la recherche d'une typologie des personnages enfants à travers la littérature enfantine. Revue de Statistique Appliquée XXI, 23–49 (1973)

[15] Lerman, I.C.: Classification et analyse ordinale des données. Dunod, Paris (1981)

[16] Lerman, I.C.: Justification et validité statistique d'une échelle [0,1] de fréquence mathématique pour une structure de proximité sur un ensemble de variables observées. Publications de l'Institut de Statistique des Universités de Paris 29, 27–57 (1984)

[17] Lerman, I.C.: Conception et analyse de la forme limite d'une famille de coefficients statistiques d'association entre variables relationnelles. Mathématiques et Sciences Humaines 118, 33–52 (1992)

[18] Lerman, I.C.: Analyse de la vraisemblance des liens relationnels: une méthodologie d'analyse classificatoire des données. In: Bennani, Y., Viennet, E. (eds.) Apprentissage Artificiel et Fouille de Données 2009. RNTI, vol. RNTI A3, pp. 93–126. Cépaduès (2009)

[19] Lerman, I.C.: Facets of the set theoretic representation of categorical data. Publication Interne 1988, IRISA-INRIA (2012)

[20] Lerman, I.C., Azé, J.: A new probabilistic measure of interestingness for association rules, based on the likelihood of the link. In: Quality Measures in Data Mining 2007. SCI, vol. 43, pp. 207–236. Springer, Heidelberg (2007)

[21] Lerman, I.C., Gras, R., Rostam, H.: Élaboration et évaluation d'un indice d'implication pour des données binaires i et ii. Mathématiques et Sciences Humaines, 74–75, 5–35, 5–47 (1981)

[22] Lerman, I.C., Guillaume, S.: Analyse comparative d'indices d'implication discriminants fondés sur une échelle de probabilité. Rapport de Recherche, INRIA, Rennes, 7187, Février, 85 pages (2010)

[23] Morineau, A., Rakotomalala, R.: Critère VT100 de sélection des règles d'association. In: Ritschard, G., Djeraba, C. (eds.) Actes de Extraction et Gestion de Connaissances, EGC 2006. RNTI, pp. 581–592. Cépaduès (2006)

[24] Piatetsky-Shapiro, G.: Discovery, analysis, and presentation of strong rules. In: Knowledge Discovery in Databases 1991, pp. 229–248. MIT Press (1991)

[25] Rabaseda, S., Rakotomalala, R., Sebban, M.: Discretization of continuous attributes: a survey of methods. In: Proceedings of the Second Annual Joint Conference on Information Sciences, pp. 164–166 (1995)

[26] Rakotomalala, R., Morineau, A.: The TVpercent principle for the counterexamples statistic. In: Gras, R., Suzuki, E., Guillet, F., Spagnolo, F. (eds.) Statistical Implicative Analysis, pp. 449–462. Springer (2008)

[27] Ritschard, G.: De l'usage de la statistique implicative dans les arbres de classification. In: Gras, R., et al. (eds.) Analyse Statistique Implicative, pp. 305–316. Troisième Rencontre Internationale (2005)

[28] Tan, P.N., Kumar, V., Srivastava, J.: Selecting the right interestingness measure for association patterns. In: Proceedings of the 8th ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2002, pp. 32–41. ACM (2002)

A New Way for Hierarchical and Topological Clustering

Hanane Azzag and Mustapha Lebbah

1 Introduction

Clustering is one of the most important unsupervised learning problems. It deals with finding a structure in a collection of unlabeled data points. Hierarchical clustering algorithms are typically more effective in detecting the true clustering structure of a structured data set than partitioning algorithms. We find in literature several important research in hierarchical cluster analysis [Jain et al., 1999]. Hierarchical methods can be further divided to agglomerative and divisive algorithms, corresponding to bottom-up and top-down strategies, to build a hierarchical clustering tree. Another works concerning hierarchical classifiers are presented in [Jiang et al., 2010]. In this paper we propose a new way to build a set of self-organized hierarchical trees.

Self-organizing models (SOM) are often used for visualization and unsupervised topological clustering [Kohonen et al., 2001]. They allow projection in small spaces that are generally two dimensional. Some extensions and reformulations of SOM model have been described in the literature [Hammer et al., 2009], [Bishop et al., 1998, Rossi and Villa-Vialaneix, 2010]. Hierarchical version of SOM are also defined in [Vesanto and Alhoniemi, 2000]. A variety of these topological maps algorithms are derived from the first original model proposed by Kohonen. All models are different from each other but share the same idea: depict large datasets on a simple geometric relationship projected on a reduced topology (2D). SOM model has several tree-structured versions such as TS-SOM [Koikkalainen and Horppu, 2007], GH-SOM [Dittenbach et al., 2000], TreeSOM

Hanane Azzag · Mustapha Lebbah
Université Paris 13, Sorbonne Paris Cité,
Laboratoire d’Informatique de Paris-Nord (LIPN),
CNRS(UMR 7030),
99, avenue Jean-Baptiste Clément
Villetaneuse, 93430 France
e-mail: firstname.secondname@lipn.univ-paris13.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 85–97.
DOI: 10.1007/978-3-642-35855-5_5

[Samsonova et al., 2006] and SOM-AT [Peura, 1998]. Our approach should not be confused with these methods, it is totally different from TS-SOM, GH-SOM in which the map architecture has the form of a tree. Each neuron of map now becomes one node of tree. On the other hand TreeSOM proposed to generate a hierarchical tree where only the leaf nodes may get many data elements, and other nodes none. SOM-AT introduce matching and adjusting schemes for input data attribute trees. The most optimal tree is selected to represent input data.

Concerning the visual aspect of our studies, we can find in the literature several algorithms for visualizing hierarchical structures, which are mostly 2D. One may cite treemap method which recursively maps the tree to embedded rectangles [Johnson and Shneiderman, 1991, Shneiderman, 1992]. Hyperbolic displays have also been studied in 2D and 3D [Carey et al., 2003]. Another example is the cone tree [Robertson et al., 1991]: the root of the tree is the top of a cone. The subtrees of this root are all included in this cone. The size of a cone may depend on the number of nodes which are present in each tree. In this work we introduce a new method named MTM (Map Tree Map), that proposes a self-organizing treemap, which provides a simultaneously hierarchical and topological clustering. Each cell of map represents a tree structured data and treemap method provides a global view of the local hierarchical organization. Data moves toward a map of trees according to autonomous rules that are based on nearest neighborhood approach. The topological process of the proposed algorithm is inspired from SOM model and the rules for building tree are inspired from autonomous artificial ants method [Azzag et al., 2007, Slimane et al., 2003]. The rest of this paper is organized as follows: in section 2, we present both SOM model and proposed model. In section 3, we show the experimental results on several data sets. These data sets illustrate the use of this algorithm for topological and visual hierarchical clustering. Finally we offer some concluding comments of proposed method and the further research.

2 Hierarchical and Topological Clustering Model

We present in this paper a new model that provides a hierarchical clustering of data where each partition is a forest of trees organized in a 2D map. The obtained map is inspired from SOM algorithm and could be seen as a forest of trees.

2.1 Self-Organizing Maps

Self-organizing maps are increasingly used as tools for visualization and clustering, as they allow projection over small areas that are generally two dimensional. The basic model proposed by Kohonen (SOM: Self-Organizing-Map) consists of a discrete set C of cells called map. This map has a discrete topology defined by undirected graph, it is usually a regular grid in 2 dimensions. We denote p as the number of cells. For each pair of cells (c,r) on the map, the distance $\delta(c,r)$ is defined as the length of the shortest chain linking cells r and c on the grid without sub-trees. For each cell c this distance defines a neighbor cell; in order to control the neighborhood area, we introduce a kernel positive function $\mathcal{K}$ ($\mathcal{K} \geq 0$ and $\lim_{|x|\to\infty} \mathcal{K}(x) = 0$). We define the mutual influence of two cells c and r by $\mathcal{K}(\delta(c,r))$. In practice, as for traditional topological map we use smooth function to control the size of the neighborhood as:

$$\mathcal{K}(\delta(c,r)) = \exp\left(\frac{-\delta(c,r)}{T}\right) \tag{1}$$

Using this kernel function, T becomes a parameter of the model. As in the SOM algorithm, we increase T from an initial value T_{max} to a final value T_{min}. Let $\mathcal{R}^d$ be the euclidean data space and $\mathcal{A} = \{\mathbf{x}_i; i = 1,\ldots,n\}$ a set of observations, where each observation $\mathbf{x}_i = (x_i^1, x_i^2, \ldots, x_i^d)$ is a continuous vector in $\mathcal{R}^d$. For each cell c of the grid, we associate a referent vector $\mathbf{w}_c = (w_c^1, w_c^2, \ldots, w_c^j, \ldots, w_c^d)$ of dimension d. We denote by $\mathcal{W}$ the set of the referent vectors. The set of parameter $\mathcal{W}$, has to be estimated from $\mathcal{A}$ iteratively by minimizing a cost function defined as follows:

$$\mathcal{R}(\phi,\mathcal{W}) = \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{r \in C} \mathcal{K}^T(\delta(\phi(\mathbf{x}_i),r))||\mathbf{x}_i - \mathbf{w}_r||^2 \tag{2}$$

where ϕ assigns each observation $\mathbf{x}$ to a single cell in the map C. In this expression $||\mathbf{x} - \mathbf{w}_r||^2$ is a square of the Euclidean distance. At the end of learning, SOM provides a partition of p subsets.

2.2 Proposed Model: Map Treemap

The proposed model uses the same grid process, combined with a new concept of neighborhood. Our model seeks to find an automatic clustering that provides a hierarchical and topological organization of observations $\mathcal{A}$. This model is presented as regular grid in 2D that has a topological order of p cells. Each cell c is the 'root support' of a tree denoted by $Tree_c$ and each node $N_{\mathbf{x}_i}$ of the tree represents a data $\mathbf{x}_i$. More precisely the proposed model defines a forest of trees organized on a 2D map called C. Taking into account the proximity between two trees on the map C is a useful information which allows to define a topological neighborhood relation used in traditional topological maps. Thus, for each pair of trees $Tree_c$ and $Tree_r$ on the map, the distance $\delta(c,r)$ is defined as the length of the shortest chain linking cells r and c on the map associated to $Tree_c$ and $Tree_r$. To model the influence between $Tree_r$ and $Tree_c$ we use a neighborhood function $\mathcal{K}$ defined above (eq. 1). Thus, the mutual influence between $tree_c$ and $tree_r$ is defined by the function $\mathcal{K}^T(\delta(c,r))$ where T represents the temperature function that controls the size of the neighborhood. We associate to each tree a representative point denoted $\mathbf{w}_c$ which is a given data denoted $\mathbf{x}_i$ in $tree_c$ ($\mathbf{w}_c = \mathbf{x}_i \in tree_c$). Choosing a representative point allows easily adapting our algorithm to any type of data (categorical, binary, and mixed data data . . . etc). The objective function of self-organizing trees is defined as follows:

$$\mathcal{R}(\phi,\mathcal{W}) = \sum_{\mathbf{c}\in\mathbf{C}} \sum_{\mathbf{x}_i\in Tree_c} \sum_{r\in C} \mathcal{K}^T(\delta(\phi(\mathbf{x}_i),r))||\mathbf{x}_i - \mathbf{w}_r||^2 \tag{3}$$

Minimizing the cost function $\mathcal{R}(\phi,\mathcal{W})$ is a combinatorial optimization problem. In practice, we seek to find the best (optimal) solution by using batch version. In this work we propose to minimize the cost function in the same way as "batch" version but using statistical characteristics provided by trees to accelerate the convergence of the algorithm. Three basic steps are necessary to minimize the cost function and are defined as follows:

1. **Assignment step.** Each datum $\mathbf{x}_i$ is connected in $Tree_c$ and forms a hierarchical relationship noted parent-child. We denote by *nodeChild*($\mathbf{x}_i$) the function, which provides all child nodes with the same parent node $N_{\mathbf{x}_i}$ associated to the data $\mathbf{x}_i$. At step $t = 0$, $nodeChild(\mathbf{x}_i) = \mathbf{x}_i$.

   Assignment step consists of finding for each observation $\mathbf{x}_i$ a best match tree called "Winner" using the assignment function named χ. This tree is also defined as winner tree. The children nodes of $\mathbf{x}_i$ ($nodeChild(\mathbf{x}_i)$). In other words, all nodes of tree $N_{\mathbf{x}_i}$ are recursively assigned to the winning tree. The assignment function is defined as follows:

$$\chi(nodeChild(\mathbf{x}_i)) = \arg\min_r \sum_{c\in C} \mathcal{K}^T\,(\delta(r,c))\,||\mathbf{x}_i - \mathbf{w}_c||^2 \tag{4}$$

   where, $\mathbf{w}_c \in \mathcal{A}$

2. **Building Tree step.** In this step we seek to find the best position of a given data $\mathbf{x}_i$ in the $Tree_c$ associated to cell c. We use connections/disconnections rules inspired from [Azzag et al., 2007, Slimane et al., 2003]. Each data will be connected to its nearest neighbor. The particularity of the obtained tree is that each node N whether it is a leaf or an internal node represents a given data $\mathbf{x}_i$. In this case, $N_{\mathbf{x}_i}$ denotes the node that is associated to the data $\mathbf{x}_i$, $N_{\mathbf{x}_{pos}}$ represents current node of the tree and $N_{\mathbf{x}_{i+}}$ the node connected to $N_{\mathbf{x}_{pos}}$, which is the most similar (closest by distance) to $N_{\mathbf{x}_i}$. We also note V_{pos} the local neighborhood observed by $N_{\mathbf{x}_i}$ and the node connected $N_{\mathbf{x}_{pos}}$ in the concerned tree.

   Let $T_{Dist}(N_{\mathbf{x}_{pos}})$ be the highest distance value which can be observed among the local neighborhood V_{pos}. $\mathbf{x}_i$ is connected to $N_{\mathbf{x}_{pos}}$ if and only if the connection of $N_{\mathbf{x}_i}$ further increases this value. Thus, this measure represents the value of the maximum distance observed in the local neighborhood V_{pos}, between each pair of data connected to the current node $N_{\mathbf{x}_{pos}}$:

$$\begin{aligned} T_{Dist}(N_{\mathbf{x}_{pos}}) &= Max_{j,k}\left\|N_{\mathbf{x}_j} - N_{\mathbf{x}_k}\right\|^2 \\ &= Max_{j,k}\left\|\mathbf{x}_j - \mathbf{x}_k\right\|^2 \end{aligned} \tag{5}$$

   Connections rules consist of comparing a node $N_{\mathbf{x}_i}$ to the nearest node $N_{\mathbf{x}_{i+}}$. In the case where both nodes are sufficiently far away ($||N_{\mathbf{x}_i} - N_{\mathbf{x}_{i+}}||^2 > T_{Dist}(N_{\mathbf{x}_{pos}})$)

the node $N_{\mathbf{x}_i}$ is connected to its current position $N_{\mathbf{x}_{pos}}$. Otherwise, the node $N_{\mathbf{x}_i}$ associated to $\mathbf{x}_i$ is moved toward the nearest node $N_{\mathbf{x}_{i+}}$. Therefore, the value T_{Dist} decreases for each node connected to the tree. In fact, each connection of a given data $\mathbf{x}_i$ implies a local minimization of the value of the corresponding T_{Dist}. Therefore it implies a minimization of the cost function (3).

It can be observed that, for any node of the tree, the value $T_{Dist}(N_{pos})$ is only decreasing, which ensures the termination and the minimization of the cost function. At the end of the tree construction step, each cell c of the map C will be associated to $tree_c$.

3. **Representation step.**
Minimizing the cost function $\mathcal{R}(\phi, \mathcal{W})$ with respect to $\mathbf{w}_c$ corresponds to finding the point that minimizes all local distances weighted by neighborhood function.

$$\mathbf{w}_c = \min_{\mathbf{w}_c \in tree_c} \sum_{\mathbf{x}_i \in \mathcal{A}} \mathcal{K}(\delta(c, \chi(\mathbf{x}_i))) \|\mathbf{x}_i - \mathbf{w}_c\|^2, \quad \forall c \in C \tag{6}$$

The temperature T evolves according to the iterations from T_{max} to T_{min} in the same way as traditional topological maps. In the practical case we use neighborhood function as following:

$$\mathcal{K}^T(x) = e^{\frac{-\delta(r,c)}{T}}$$

We present below the detail of MTM algorithm 4.

2.3 Topological Order in *MTM* Model

The decomposition of the cost function $\mathcal{R}$ that depends on the value of T, allows to rewrite its expression as follows:

$$\mathcal{R}^T(\chi, \mathcal{W}) = \left[\sum_c \sum_{r \neq c} \sum_{\mathbf{x}_i \in tree_r} \mathcal{K}^T(\delta(c,r)) \|\mathbf{x}_i - \mathbf{w}_r\|^2\right] + \left[\mathcal{K}^T(\delta(c,c)) \sum_c \sum_{\mathbf{x}_i \in tree_c} \|\mathbf{x}_i - \mathbf{w}_c\|^2\right]$$

where $\delta(c,c) = 0$

The cost function $\mathcal{R}$ is decomposed into two terms. In order to maintain the topological order between trees, minimizing the first term will bring trees corresponding to two neighboring cells. Indeed, if $Tree_c$ and $Tree_r$ are neighbors on the map, the value of $\delta(c,r)$ is lowest and in this case the value of $\mathcal{K}^T(\delta(c,r))$ is the highest. Thus, minimizing the first term has as effect reducing the value of the cost function. Minimizing the second term corresponds to the minimization of the inertia of points connected to the $Tree_c$ (in the case of Euclidean space). Minimizing this term is considered as applying hierarchical clustering algorithm (AntTree).

Algorithm 4. Detail of *MTM* algorithm

```
Input: Map $C$ of $n_c$ cells, learning set $A$, the number of iteration $n_{iter}$
Output: Map $C$ of $n_c$ empty cells or which contain sub-tree
for $c \in C$ do
    $\mathbf{w}_c = \mathbf{x}_i$
end for
{ /* random Initialization of the map */}
for $t = 1$ to $n_{iter}$ do
    for $\mathbf{x}_i \in A$ do
        if first assignment of $\mathbf{x}_i$ then
            Find the “wining” cell $\chi(\mathbf{x}_i)$ by using the assignment function defined in (eq. 4)
            Associate the data $\mathbf{x}_i$ to a node $N_{\mathbf{x}_i}$,
            Connect the node $N_{\mathbf{x}_i}$ in the sub-tree $Tree_{\chi(\mathbf{x}_i)}$ by using connection rules
            Update the representative point $\mathbf{w}_c$ by using the defined expression (eq. 6)
        else
            Find the “wining” cell $c_{new} = \chi(nodechild(\mathbf{x}_i))$ by using function defined in 4
            {/* $t^{th}$ assignment for the data $\mathbf{x}_i$*/}
        end if
        if $c_{new} \neq c_{old}$ then
            Assign data $\mathbf{x}_i$ and the child node $nodechild(\mathbf{x}_i)$ to the new cell $c_{new}$
            Connect the node $N_{\mathbf{x}_i}$ and the child node in the sub-tree $tree_{c_{new}}$ by using connection rules.
            Update the two representative points $\mathbf{w}_{c_{old}}$ and $\mathbf{w}_{c_{new}}$ by using the defined function (eq.6)
        end if
    end for
end for
```

For different values of temperature T, each term of the cost function has a relative relevance in the minimization process. For large values of T, the first term is dominant and in this case, the priority is to preserve the topology. When value of T is lowest, the second term is considered in the cost function. In this case, the priority is to determine representative compact trees. Our model provides a solution to regularized AntTree algorithm: regularization is achieved through the constraint of ordering on the trees.

3 Comparatives Results

3.1 Visual Exploration of MTM

We have tested and compared the proposed algorithm on several datasets that have been generated with Gaussian and Uniform distributions. Others have been extracted from the machine learning repository [Blake and Merz, 1998] and have several difficulties (fuzzy clustering, no relevant feature, ...). Before comparing our numerical results, we present a map visualization with associated treemaps.

Treemap is a visualization technique introduced in [Shneiderman, 1992]. An important feature of treemaps is that it makes very efficient use of display space. Thus it is possible to display large trees with many hierarchical levels in a minimal amount of space (2D). Treemap can be helpful when dealing with large clustered tree. Treemaps lend themselves naturally to showing the information encapsulated in the clustering tree. Viewing a tree at some level of abstraction, the viewer is really looking at nodes belonging to some level in the tree. A treemap can display the whole structure of trees and allow the users to place the current view in context. In the proposed visualization technique, each tree is represented by a treemap. This aims to obtain an automatic organization of treemaps on a 2D map. Figure 1 shows an example of four tree structures with its corresponding treemaps. The positioning of tree nodes in a treemap is a recursive process. The nodes are represented as rectangles of various shapes. First, the children of the root are placed across the display area horizontally. Then, for each node N already displayed, each of N's children is placed across vertically within N's display area. This process is repeated, alternating between horizontal and vertical placement until all nodes have been displayed. We note that each rectangle is colored according to the real label of its corresponding node/data. This makes easy a visual comparison of homogeneous clusters.

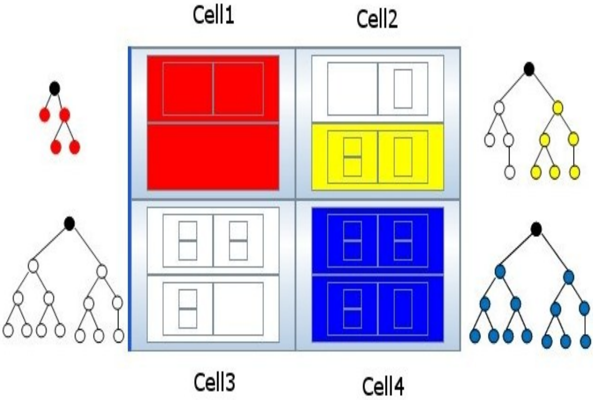


Fig. 1 Map treemaps representation: 2 × 2 MTM

In figure 1 each treemap represents a hierarchical organization of data belonging to cluster "tree". Thus, MTM approach has several properties that allow obtaining a simultaneous topological hierarchical clustering. We observe in figure 1 that data placed in the $tree_c$ are similar to $N_{\mathbf{x}_i}$ and the child nodes of $N_{\mathbf{x}_i}$ represent recursively subtrees that are dissimilar to their "sister" subtrees. In order to best analyze the obtained result, we have learned for each dataset 1 × 1 MTM in order to build a single treemap. Figures 2,3, and 4 display some example of 1 × 1 MTM and 4 × 4 MTM. Observing both maps on each dataset, we find that our algorithm provides a MTM, which is a multi-divisions of the main treemap. We can see that topological and hierarchical organization of data is more apparent. In order to visualize the coherence between intra-organization of treemaps and the label points, we assign one color to each label. In each figure (2, 3, 4), we distinguish two regions on the MTM that are dedicated to the pure and mixed clusters. Map presented in Figure 2.b shows diagonal from right to left is dedicated to one class (colored in blue) and the treemap positioned in the bottom right is a mixed cluster. We observe in this treemaps, that yellow point is positioned in a lower level on the tree, this behavior is normal since the yellow classes are situated in the neighborhood. Same remarks concern Lsun and Tetra dataset. In figure 4 observing the top right treemap (cell) and the bottom left, we can conclude on the level and the side where cluster will become mixed. Thus, this visual analysis is done using only 2D visualization unlike SOM method where we can not conclude on which level data is positioned. This visual system allows analyst to easily navigate trough the databases and to let the user easily interact with the data and perceive details, global context and shape of the tree.

3.2 Comparison with Other Clustering Methods

We remind here that MTM model provides more information than traditional hierarchical models, K-means or others. In this work we compare the obtained result with SOM model. In this case we adopt the same parameter: map size, initial and final neighborhood.

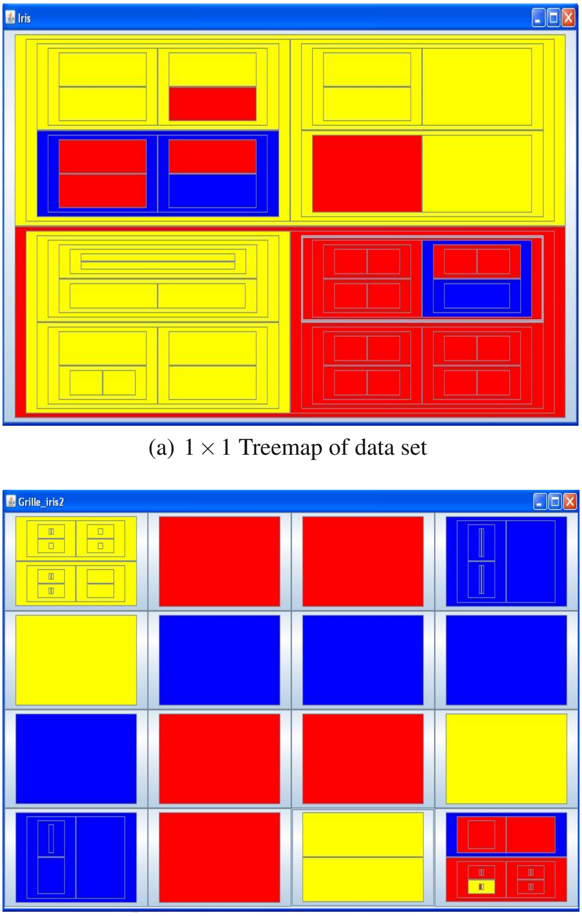
To measure the quality of map clustering, we adopt the approach of comparing results to a ‘ground truth’. We use two criterions for measuring the clustering results. The first one is Rand index which measures the percentage of observation

Table 1 Competitive results obtained with AHC, MTM and SOM using the same parameter (map size, initial and final parameter T). *DB* is the Davides-Bouldin index.

Datasets	size.	MTM		SOM		CAH	
	DB	Rand	*DB*	Rand	*DB*	Rand	
Atom(2)	800	1.4	0.88	1.47	0.51	0.81	0.77
Anneaux (2)	1000	0.80	0.61	0.90	0.51	0.50	0.55
ART1(4)	400	0.98	0.81	0.85	0.81	0.79	0.88
Demi-cercle(2)	600	0.58	0.60	0.67	0.5	0.55	0.48
Glass(7)	214	1.56	0.70	2	0.65	0.65	0.72
Hepta(7)	212	0.92	0.92	0.85	0.93	0.35	1.00
Iris(3)	150	1.06	0.75	1.03	0.75	0.43	0.77
Lsun(3)	400	0.97	0.71	1.09	0.72	0.54	0.85
Pima(2)	768	1.09	0.5	2.23	0.43	0.65	0.56
Target(6)	770	1.4	0.85	1.17	0.58	0.44	0.81
Tetra(4)	400	0.82	0.81	1.25	0.76	0.71	0.99
TwoDiamonds(2)	800	0.86	0.60	0.81	0.51	0.57	1.00

pairs belonging to the same class and which are assigned to same cluster of the map [Saporta and Youness, 2002]. The second index is Davides Bouldin criterion which [Davies and Bouldin, 1979] is used to determine the optimal number of centroids for K-means.

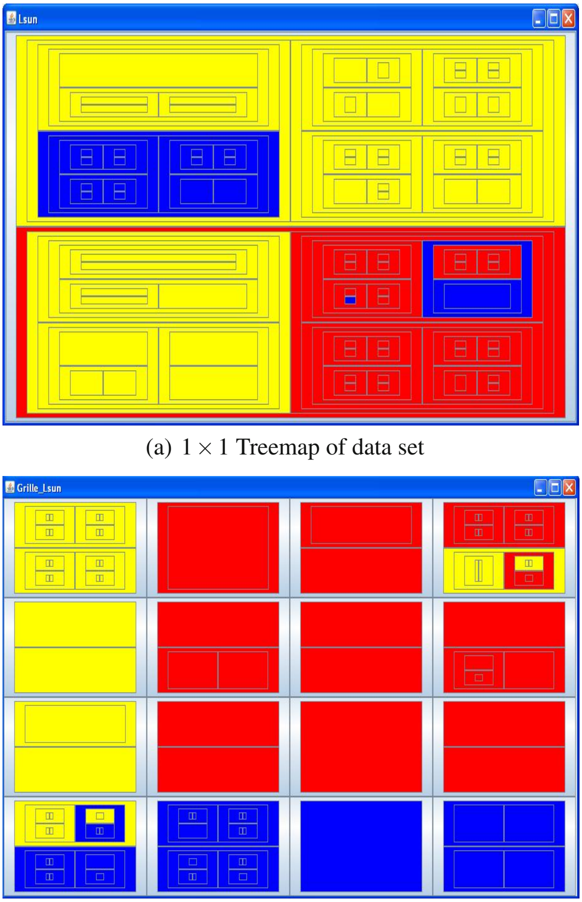
Table 1 reports clustering evaluation criterion obtained with MTM, SOM and AHC. MTM method provides results quite comparable to those obtained with SOM method on the majority of cases. Looking to columns (DB and Rand index) associated to MTM, we observe that DB index value is lower using our algorithm and rand index is highest near one for the majority of datasets.



(a) 1×1 Treemap of data set

(b) 4×4 MTM

Fig. 2 Iris Dataset



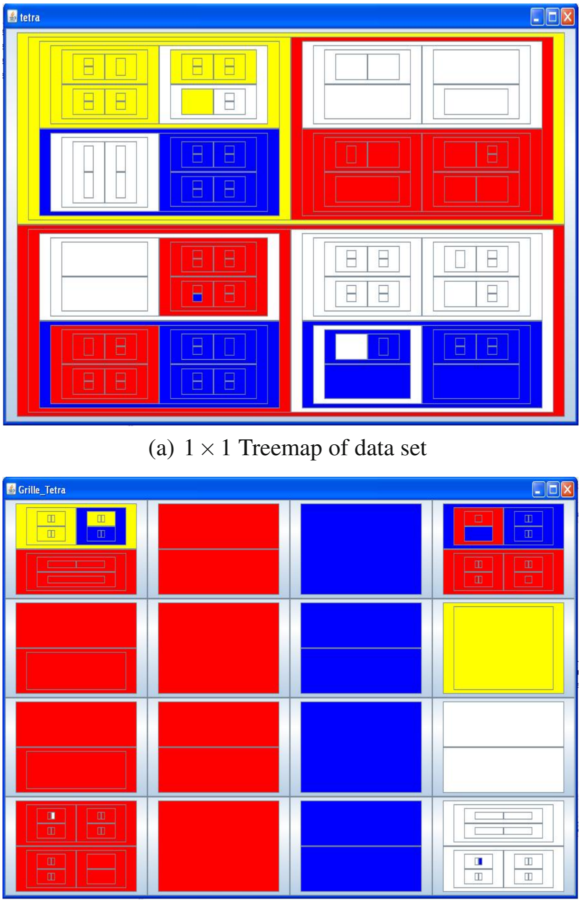
(a) 1 × 1 Treemap of data set

(b) 4 × 4 MTM

Fig. 3 Lsun Dataset

Concerning AHC method [Jain et al., 1999], we have used DB index to select the number of clusters. This justifies the best results of DB index obtained by AHC comparing to MTM. Indeed DB is lower for the majority of cases but not far away comparing to DB index obtained by MTM. Concerning Rand index values, MTM obtains similar results as AHC for the majority of cases.

Our purpose through this comparison is not to assert that our method is the best, but to show that MTM method can obtain quite the same good results as SOM or other well known clustering algorithms. Unlike SOM method or AHC, MTM does not require a posterior processing to analyze the structure of data belonging to clusters. MTM provides simultaneously hierarchical and topological



(a) 1 × 1 Treemap of data set

(b) 5 × 5 Map-Treemaps

Fig. 4 Tetra Dataset

clustering which is more interesting for visualization task. Thus MTM has two main advantages:

1. Complexity: we reduce the number of assignments. When an observation is reassigned to another tree, the entire sub-trees associated to this observation will follow it into the new cell (see expression 4).
2. Data projection and rapid visualization: In our model, we don't need to use a traditional projection of the map to get an idea about the structure of data. Treemap organization of data presents a local structure for each cell of the map.

4 Conclusion and Perspectives

In this paper, we have presented a new algorithm dedicated to hierarchical clustering that has the following properties: it provides a local hierarchical clustering of data, that allows a better visualization of the obtained organization. It generates both 2D self-organization of the trees associated to each cell and hierarchical organization provided by tree. The obtained results have been compared to those obtained by traditional Kohonen algorithm (SOM) and AHC. This comparison shows that the proposed approach is promising and can be used in various applications of data mining. The major benefits of MTM approach are the following: MTM uncovers the hierarchical structure of the data allowing the user to understand and analyze large amounts of data. Using the various emerging trees at each cell being rather small in size, it is much easier for the user to keep an overview of the various clusters.

Results presented in this papaer are preliminary and much work still be done. It is obvious that using trees for data clustering greatly speeds up the learning process, we wish to generalize these algorithms to other kind of structures which may not be trees. The same principles seem to be applicable also to graphs. Also, it will be necessary to focus on the visual aspect of our approach. Indeed, we will develop a 2D/3D view of the different trees that result from the hierarchical clustering in order to allow an interactive exploration of data.

References

[Azzag et al., 2007] Azzag, H., Venturini, G., Oliver, A., Guinot, C.: A hierarchical ant based clustering algorithm and its use in three real-world applications. European Journal of Operational Research 179(3), 906–922 (2007)

[Bishop et al., 1998] Bishop, C.M., Svensén, M., Williams, C.K.I.: Gtm: The generative topographic mapping. Neural Computation 10(1), 215–234 (1998)

[Blake and Merz, 1998] Blake, C., Merz, C.: UCI repository of machine learning databases (1998), http://www.ics.uci.edu/~mlearn/MLRepository.html

[Carey et al., 2003] Carey, M., Heesch, D.C., Rüger, S.M.: Info navigator: A visualization tool for document searching and browsing. In: Proc. of the Intl. Conf. on Distributed Multimedia Systems (DMS), pp. 23–28 (2003)

[Davies and Bouldin, 1979] Davies, D.L., Bouldin, D.W.: A cluster separation measure. IEEE Transactions on Pattern Recognition and Machine Intelligence 1(2), 224–227 (1979)

[Dittenbach et al., 2000] Dittenbach, M., Merkl, D., Rauber, A.: The growing hierarchical self-organizing map, pp. 15–19. IEEE Computer Society (2000)

[Hammer et al., 2009] Hammer, B., Hasenfuss, A., Rossi, F.: Median Topographic Maps for Biomedical Data Sets. In: Biehl, M., Hammer, B., Verleysen, M., Villmann, T. (eds.) Similarity-Based Clustering. LNCS, vol. 5400, pp. 92–117. Springer, Heidelberg (2009)

[Jain et al., 1999] Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Computing Surveys 31(3), 264–323 (1999)

[Jiang et al., 2010] Jiang, W., Raś, Z.W., Wieczorkowska, A.A.: Clustering Driven Cascade Classifiers for Multi-indexing of Polyphonic Music by Instruments. In: Raś, Z.W., Wieczorkowska, A.A. (eds.) Advances in Music Information Retrieval. SCI, vol. 274, pp. 19–38. Springer, Heidelberg (2010)

[Johnson and Shneiderman, 1991] Johnson, B., Shneiderman, B.: Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In: Proceedings of the 2nd Conference on Visualization 1991, VIS 1991, pp. 284–291. IEEE Computer Society Press, Los Alamitos (1991)

[Kohonen et al., 2001] Kohonen, T., Schroeder, M.R., Huang, T.S. (eds.): Self-Organizing Maps, 3rd edn. Springer-Verlag New York, Inc., Secaucus (2001)

[Koikkalainen and Horppu, 2007] Koikkalainen, P., Horppu, I.: Handling missing data with the tree-structured self-organizing map. In: IJCNN, pp. 2289–2294 (2007)

[Peura, 1998] Peura, M.: The self-organizing map of trees. Neural Process. Lett. 8(2), 155–162 (1998)

[Robertson et al., 1991] Robertson, G.G., Mackinlay, J.D., Card, S.K.: Cone trees: animated 3d visualizations of hierarchical information. In: CHI 1991: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 189–194. ACM Press, New York (1991)

[Rossi and Villa-Vialaneix, 2010] Rossi, F., Villa-Vialaneix, N.: Optimizing an organized modularity measure for topographic graph clustering: A deterministic annealing approach. Neurocomput. 73, 1142–1163 (2010)

[Samsonova et al., 2006] Samsonova, E.V., Kok, J.N., Ijzerman, A.P.: Treesom: Cluster analysis in the self-organizing map. neural networks. American Economic Review 82, 1162–1176 (2006)

[Saporta and Youness, 2002] Saporta, G., Youness, G.: Comparing two partitions: Some proposals and experiments. In: Proceedings in Computational Statistics, pp. 243–248. Physica Verlag (2002)

[Shneiderman, 1992] Shneiderman, B.: Tree visualization with tree-maps: A 2-D space-filling approach. ACM Transactions on Graphics 11(1), 92–99 (1992)

[Slimane et al., 2003] Slimane, A.M., Azzag, H., Monmarché, N., Slimane, M., Venturini, G.: Anttree: a new model for clustering with artificial ants. In: IEEE Congress on Evolutionary Computation, pp. 2642–2647. IEEE Press (2003)

[Vesanto and Alhoniemi, 2000] Vesanto, J., Alhoniemi, E.: Clustering of the self-organizing map. IEEE Transactions on Neural Networks 11(3), 586–600 (2000)

Summarizing and Querying Logs of OLAP Queries

Julien Aligon, Patrick Marcel, and Elsa Negre

Abstract. Leveraging query logs benefits the users analyzing large data warehouses with OLAP queries. But so far nothing exists to allow the user to have concise and usable representations of what is in the log. In this article, we present a framework for summarizing and querying OLAP query logs. The basic idea is that a query summarizes another query and that a log, which is a sequence of queries, summarizes another log. Our formal framework includes a language to declaratively specify a summary, and a language for querying and manipulating logs. We also propose a simple measure based on precision and recall, to assess the quality of summaries, and two strategies for automatically computing log summaries of good quality. Finally we show how some simple properties on the summaries can be used to query the log efficiently. The framework is implemented using the Mondrian open source OLAP engine. Its interest is illustrated with experiments on synthetic yet realistic MDX query logs.

1 Introduction

It is becoming accepted that leveraging query logs would help the user analyzing large databases or data warehouses [Khoussainova et al., 2009]. As a clear evidence of this, it has recently been shown that browsing and querying logs actually speeds up the query formulation, by supporting better query reuse [Khoussainova et al., 2011].

Julien Aligon · Patrick Marcel
Université François Rabelais Tours, Laboratoire d'Informatique, France
e-mail: firstname.lastname@univ-tours.fr

Elsa Negre
Université Paris-Dauphine, LAMSADE, France
e-mail: elsa.negre@dauphine.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 99–124.
DOI: 10.1007/978-3-642-35855-5_6

This is particularly relevant in a collaborative context for instance to issue recommendations [Chatzopoulou et al., 2009, Giacometti et al., 2009], [Giacometti et al., 2011, Stefanidis et al., 2009]. But to the best of our knowledge, even the simple problem of providing the end user with a concise representation of what is inside a large log has rarely been addressed, apart from helping a DBA to tune the RDBMS [Khoussainova et al., 2009].

Using such a summary, that avoids overwhelming the user, would have many advantages, including:

- allowing a decision maker to have a rough idea of the queries launched by other decision makers,
- helping the user to access the precise part of the log containing particular queries he/she is looking for,
- helping an administrator to manage and tune the OLAP server, e.g., if the summary indicates the frequently accessed members,
- aassisting the decision maker to perform new analysis sessions by considering the previous queries.

In this article we present and develop the work initiated in [Aligon et al., 2010, Aligon et al., 2011]. In these papers, we proposed a framework for summarizing an OLAP query log, and we studied basic properties of the framework for helping the user to query the log. The present article provides a detailed presentation of the framework and introduces its implementation as a system for summarizing and querying log files. To this end, we extend the search facilities introduced in [Aligon et al., 2011] to obtain a declarative language with which complex queries over a log file can be expressed.

Our approach is based on the idea that a log, which is a sequence of queries, is summarized by another sequence of queries, i.e., by another (much shorter) log. It entails that a query summarizes other queries. Our framework includes:

- A language tailored for OLAP queries, named *QSL*, for declaratively expressing summaries. This language is composed of binary and unary operators that allow to summarize queries.
- A greedy algorithm using *QSL* for automatically constructing summaries of query logs.
- A quality measure adapted from the classical precision and recall, that allows to measure how faithful the constructed summaries are.
- Two sub-languages of *QSL* whose properties w.r.t. the quality measure are used to ensure that summaries can help query the log efficiently.
- Compositional search operators with which the user can query the log for particular OLAP queries.

This paper is organized as follows. Next section motivates the approach with a toy example. The *QSL* query language which is at the core of our framework is presented in Section 3. Section 4 describes the quality measure based on precision and recall, that is used to assess the summaries expressed in *QSL*. In Section 5, we present the algorithm that automatically constructs summaries based on *QSL* and the quality measure. We also introduce the properties of the summaries constructed with sub-languages of *QSL*. Section 6 presents the language for querying logs, and describes how properties of the framework can be used to ensure efficient searches. Section 7 describes the implementation of the framework and the experiments conducted to evaluate its effectiveness. Section 8 discusses related work. We conclude and draw perspectives in Section 9.

2 Motivating Example

In this section, we illustrate with a toy example our approach for summarizing a log of OLAP queries. The context of this example is that of a user navigating a data warehouse. In our example, the data warehouse records sales of beverages in different locations at different times. The dimensions of this data warehouse are given in Figure 1. Consider a sequence of queries $L = \langle q_1, q_2, q_3 \rangle$ where q_1 is the first query launched, q_2 the second one and q_3 the last one. Suppose these queries are logged in a log L and ask respectively for:

1. The sales of Pepsi and Coke for July 2008, in cities Paris or Marseille,
2. The sales of Coke for July 2008, in regions North or South,
3. The sales of Orangina for the second semester 2008, in regions North or South.

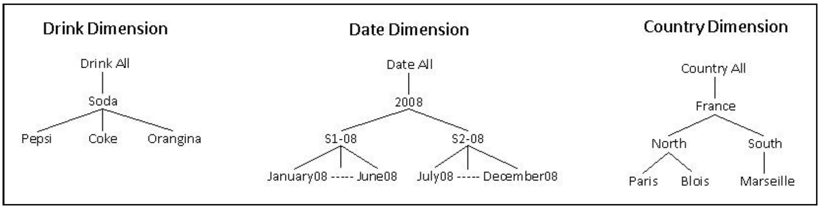


Fig. 1 Dimensions used in the toy example

Assume we want to summarize these queries by another query. Various solutions are possible. First, we can summarize the queries by retaining for each dimension the most frequent members. This could be of interest for a DBA who would like to know what indices to store. In that case, the resulting query would ask for sales of Coke in regions North or South during July 2008 (i.e., query q_2).

A second alternative would be to summarize the queries with another query having for each dimension the members that cover all members present in the initial queries. For example, note that Pepsi, Coke and Orangina are sodas, cities Paris and Marseille and regions North and South are in France and all three queries concern year 2008. The query summarizing the log L would then ask for the sales of Soda in France in 2008. The user interested in more details on the query could then query the log to find the queries that were indeed launched. Finally, note that we can have a compromise by summarizing q_1 and q_2 first, say with the second alternative, and then summarizing the resulting summary with q_3, say with the first alternative. In that case, we would obtain the query asking for the sales of Soda and Orangina in France, region North and region South, for year 2008 and the second semester of 2008.

These examples show the need for flexibility in how the summary is computed. This is why in our approach we propose that summaries can be specified declaratively with a query manipulation language called *QSL*. *QSL* expressions are used to combine several queries into a query that summarises them. Note that so far, we have illustrated the problem of summarizing queries by another query. But a set of queries could be summarized by another set of queries. Moreover, summaries for a log should respect the fact that logs are sequences of queries. For instance, consider again L, this log could be summarized by the sequence $\langle q'_1, q_3 \rangle$ where q'_1 is a summary of q_1 and q_2 asking for the sales of Soda in France in the second semester of 2008. To automatically construct a summary from a log, we propose an algorithm that constructs *QSL* expressions for summarising subsequences of the log.

In addition, as various summaries can be computed from one log, the quality of these summaries should be evaluated. For instance, for our first alternative, the quality measure should take into account the fact that 'Orangina' is present in the log but not in the summary. In our second alternative, this measure should take into account that indeed 'North' and 'South' covers 'Paris' and 'Marseille' but also 'Blois', that is not present in the log. We propose such a quality measure that extends the classical notions of precision and recall.

Finally, note that summaries computed from a log may not give precise information on the queries in the log. For instance the user may wish to know if a query on member 'Blois' appears in the log, what are all the queries of the log that deal with 'drink' or one of its descendant, or what are the queries in the log following queries dealing with 'Coke'. We thus propose two operators that allow to express such searches on a log (or even on a summary).

3 *QSL*: A Query Summarizing Language

In this section, we formally define the manipulation language, called *QSL*, used to summarize OLAP queries.

3.1 Preliminary Definitions

As the query summarizing language is tailored for OLAP queries, we first begin with the definition of an OLAP query. Note that in this paper, we do not consider query result, and thus the definition of a query result is not given.

An n-dimensional cube $C = \langle D_1, \ldots, D_n, F \rangle$ is defined as the classical $n+1$ relation instances of a star schema, with one relation instance for each of the n dimensions D_i and one relation instance for the fact table F. For a dimension D_i having schema $S_i = \{L^i_1, \ldots, L^i_{d_i}\}$, a member m is any constant in $\bigcup_{L^i_j \in S_i} \pi_{L^i_j}(D_i)$. For a dimension D_i, we consider that members are arranged into a hierarchy $<_i$ and we note $m <_i m'$ (or $m < m'$ or m' covers m) the fact that the member m' is the ancestor of m in this hierarchy.

Given such a cube, a cell reference (or reference for short) is an n-tuple $\langle m_1, \ldots, m_n \rangle$ where m_i is a member of dimension $D_i, \forall i \in [1,n]$. We define multidimensional queries as sets of references that can be expressed as Cartesian products of multisets. The reason for having multisets is to be able to define operators that count members' occurrences. In this work, we distinguish between a query and its expression called query expression. A query expression is a tuple of multisets, one multiset of members in each dimension. The cross-product of these multisets is a multiset of references, which is the query.

Definition 1. (Query expression and Query) Given an n-dimensional cube $C = \langle D_1, \ldots, D_n, F \rangle$, let R_i be a multiset of members of dimension $D_i, \forall i \in [1,n]$. A query expression $q = \langle R_1, \ldots, R_n \rangle$ is a tuple of multisets of members, one for each dimension D_i of C. Given such an expression, the query specified by q is the multiset of references $R_1 \times \ldots \times R_n$.

The distinction between query expression and query is needed since a query can be specified by different query expressions. For instance, the two following expressions $\langle \{a\}, \{b,b\} \rangle$ and $\langle \{a,a\}, \{b\} \rangle$ both specify query $\{\langle a,b \rangle, \langle a,b \rangle\}$. When the context is clear, a query expression and the query it specifies will be confounded.

A log L is a finite sequence of query expressions.

Definition 2. (Log) Let C be a cube and S_C be a set of queries over C. A log L of m queries over C is a function from an ordered set $pos(L)$ of integers (called positions) of size m to S_C.

A log will be noted $L = \langle q_1, \ldots, q_m \rangle$. The set of positions of a log L is noted $pos(L)$. The set of query expressions appearing in a log L is noted $queries(L)$. We note $q \in L$ for a log L if $q \in queries(L)$. In what follows, we assume an n-dimensional cube $C = \langle D_1, \ldots, D_n, F \rangle$. In the subsequent definitions, i ranges from 1 to n. For a query expression $q = \langle R_1, \ldots, R_n \rangle$, $m_i(q) = R_i$ denotes its multiset of members in dimension D_i. The multiset $m_i(q)$ will be noted $\langle S_i, f_i \rangle$, where S_i is a set and f_i is a function giving the occurrences of each element of S_i.

Example 1. Consider the three queries q_1, q_2 and q_3 of the toy example described in the previous section. Note that q_1 can be expressed in the MDX query language:
$SELECT\ \{[Drink].[DrinkAll].[Soda].[Pepsi],$
$[Drink].[DrinkAll].[Soda].[Coke]\}\ ON\ COLUMNS$
$Crossjoin(\{\ [Country].[CountryAll].[France].[North].[Paris],$
$[Country].[CountryAll].[France].[South].[Marseille]\},$
$\{[Date].[DateAll].[2008].[S2\text{-}08].[July08]\})\ ON\ ROWS$
$FROM\ SalesCube$

We have $m_1(q_1) = \{Pepsi, Coke\}$, $m_2(q_1) = \{July08\}$,
$m_3(q_1) = \{Paris, Marseille\}$. The query expression is:

$q_1 = \langle \{Pepsi, Coke\}, \{July08\}, \{Paris, Marseille\} \rangle$. The query expressions q_2 and q_3 are:

- $q_2 = \langle \{Coke\}, \{July08\}, \{North, South\} \rangle$
- $q_3 = \langle \{Orangina\}, \{S2\text{-}08\}, \{North, South\} \rangle$

The language we propose is composed of unary operators and binary operators that manipulate query expressions and output a query expression, that is called a summary query (or simply summary for short). The main idea behind the definition of these operators is that they operate dimension-wise: They define a new query expression from the one(s) in parameter by treating each dimension independently. We now present formally these operators, starting with the binary operators.

3.2 The Binary Operators of QSL

The first operators are the classical bag operators [Garcia-Molina et al., 2008] extended to multiple dimensions.

Definition 3. (Bag operators) Given two query expressions q_1 and q_2 and $op \in \{\cup_B, \cap_B, \setminus_B\}$, $q_1\ op\ q_2$ is the query expression q with $\forall i \in [1, \ldots, n], m_i(q) = m_i(q_1)\ op\ m_i(q_2)$.

Example 2. Consider the first two query expressions of Example 1, we have:

- $q_4 = q_1 \cup_B q_2 = \langle \{Pepsi, Coke, Coke\}, \{July08, July08\}, \{Paris, Marseille, North, South\} \rangle$
- $q_5 = q_1 \cap_B q_2 = \langle \{Coke\}, \{July08\}, \emptyset \rangle$
- $q_6 = q_1 \setminus_B q_2 = \langle \{Pepsi\}, \emptyset, \{Paris, Marseille\} \rangle$

Note that q_5 and q_6 are two different expressions of the same query which is the empty set.

The next operator gives priority to one query expression over the other.

Definition 4. (Priority operator) Given two query expressions q_1 and q_2. $q_1 \lhd q_2$ gives priority to q_1 over q_2. Hence, the priority operator $\lhd$ is simply defined by $q_1 \lhd q_2 = q_1$.

3.3 The Unary Operators of QSL

Our first operator outputs, for a query expression q in parameter, a query expression for which only the most frequent members of q in each dimension are retained.

Definition 5. (Mostfreq operator) Let q be a query expression with $m_i(q) = \langle S_i, f_i \rangle$ for all i. $mostfreq(q)$ is the query expression q' with $\forall i \in [1, n], m_i(q') = \langle S'_i = \{m \in S_i | \nexists m' \in S_i, f_i(m') > f_i(m)\}, f_{i_{|S'_i}} \rangle$ ($f_{i_{|X}}$ denotes the restriction of a function f_i to the set X).

Example 3. $mostfreq(q_4) = \langle \{Coke, Coke\}, \{July08, July08\}, \{Paris, Marseille, North, South\} \rangle$.

Our second operator outputs, for a query expression q in parameter, a query expression for which only the most general members of q in each dimension are retained, w.r.t. the hierarchy of the dimension.

Definition 6. (Max operator) Let q be a query expression. $max(q)$ is the query expression q' with $\forall i \in [1,n], m_i(q') = \langle S'_i = \{m \in m_i(q) | \nexists m' \in m_i(q), m <_i m'\}, f_{i_{|S'_i}} \rangle$.

Example 4. $max(q_4) = \langle \{Pepsi, Coke, Coke\}, \{July08, July08\}, \{North, South\} \rangle$.

Our last operator outputs, for a query expression q in parameter, a query expression for which only the lowest common ancestors of the members of q in each dimension are retained, w.r.t. the hierarchy of the dimension.

Definition 7. (lca operator) Let q be a query expression. Let lca be the function that outputs, for a given set of members M in dimension D_i, their common ancestor w.r.t. $<_i$, i.e., $\{m \in D_i | \forall m' \in M, (m' <_i m) \wedge \nexists m'', (m' <_i m'' \wedge m'' <_i m)\}$, or, if $lca(m) = \emptyset$ (i.e., if m is the All member) then $lca(m) = \{m\}$. Then, $lca(q)$ is the query expression q' with $\forall i \in [1, \ldots, n], m_i(q') = \langle lca(m_i(q)) \rangle$.

Example 5. $lca(q_4) = \langle \{Soda\}, \{S2\text{-}08\}, \{France\} \rangle$.

3.4 Expression of Various Summaries

We now briefly illustrate how *QSL* can be used. For instance, consider a log L composed of 3 query expressions: $L = \langle q_1, q_2, q_3 \rangle$. This log can be summarized by the query expression q_s^1 that retains only the members that appear in all queries for each dimension, i.e., $q_s^1 = q_1 \cap_B q_2 \cap_B q_3$. Alternatively, L can be summarized by taking into account the frequency of the members used in the log: $q_s^2 = mostfreq(q_1 \cup_B q_2 \cup_B q_3)$. Finally, L can be summarized by a query roughly indicating the parts of the cube that were explored: $q_s^3 = lca(q_1 \cup_B q_2 \cup_B q_3)$. We illustrate these possibilities on our running example.

Example 6. Summarizing by retaining the common members of all queries for each dimension gives: $q_s^1 = (q_1 \cap_B q_2 \cap_B q_3) = \langle \emptyset, \emptyset, \emptyset \rangle$. Summarizing basing on the frequencies of the members gives: $q_s^2 = mostfreq(q_1 \cup_B q_2 \cup_B q_3) = \langle \{Coke\}, \{July08\}, \{North, South\} \rangle$. Summarizing with lca gives: $q_s^3 = lca(q_1 \cup_B q_2 \cup_B q_3) = \langle \{Soda\}, \{2008\}, \{France\} \rangle$.

3.5 *Properties of* QSL

We first note that the *QSL* language cannot be presented as an algebra. In particular, it is neither minimal, nor complete with respect to query expressions. For instance, the intersection operator can be simulated using the difference operator, hence the non minimality. In addition, not all query expressions can be computed using *QSL* due to the fact that no operation enables to move down along hierarchies. Achieving minimality and completeness, though theoretically compelling, may be of little practical use. For instance, it is well known that dropping minimality enables dedicated optimisations, as it is the case for outer-join in the relational algebra. Nevertheless, in the case of *QSL*, minimality can be achieved by dropping intersection. As to completeness, instead of defining other operators, *QSL* completeness can be characterized with respect to the kind of query expressions it can compute, which are more general expressions (in the sense of Definition 10, introduced in Section 6.2). While a precise characterization is part of our future work, we list below the properties of the *QSL* operators. Some of these properties, like for instance the distributivity of *max* or the commutativity of *max* and *lca* are used in our implementation of the framework.

Let q, q_1, q_2 be query expressions. We have the following:

- $\cup_B, \cap_B, \setminus_B$ keep their classical properties [Garcia-Molina et al., 2008].
- *max* and *mostfreq* are idempotents: $max(max(q)) = max(q)$ and $mostfreq(mostfreq(q)) = mostfreq(q)$.
- *max* is distributive over $\cup_B$ and $\setminus_B$: $max(q_1 \cup_B q_2) = max(max(q_1) \cup_B max(q_2))$ and $max(q_1 \setminus_B q_2) = max(q_1 \setminus_B max(q_2))$.
- $\lhd$ is associative: $q \lhd (q_1 \lhd q_2) = (q \lhd q_1) \lhd q_2 = q$.
- *max* and *lca* commute: $max(lca(q)) = lca(max(q)) = lca(q)$.
- $mostfreq(lca(q)) = lca(q)$.

4 Assessing the Quality of a Summary

In this section, we present the measure used to evaluate the quality of summaries. We begin with an intuitive presentation, then give the formal definition and we finally give the properties of the *QSL* operators w.r.t. this measure.

4.1 Intuition

The measure should assess to which extends a query (respectively, a log), which is a set of references (respectively, of queries), is a faithful summary of another query (respectively, another log). The operators of *QSL* define summaries by adding or removing references to their operands. For instance the *lca* operator summarizes by adding references containing ancestors. The measure should thus assess the proportion of what is added or removed to define the summary. This is achieved by adapting the classical notion of precision and recall. In our context, these measures should be extended to take into account the cover relation used by the operators.

For instance, in Example 5, the expression $lca(q_1 \cup_B q_2)$ summarizes q_1 and q_2 by $\langle\{Soda\}, \{S2\text{-}08\}, \{France\}\rangle$, which specifies the query $q = \{Soda\} \times \{S2\text{-}08\} \times$

$\{France\}$. Looking at the references of q_1, q_2 and q, it can be seen that q is obtained by removing references $\{Coke, Pepsi\} \times \{July08\} \times \{Paris, Marseille, North, South\}$ and adding the reference $\{Soda\} \times \{S2\text{-}08\} \times \{France\}$. If we apply the classical precision and recall measures to evaluate its quality, both are null. However, we can consider this summary as a good summary with a good quality since the added reference covers the removed references. Its recall would then be 1 and its precision would depend on the number of references covered by the added reference and not in the removed references.

We propose to extend recall and precision by taking into account a cover relation between the elements of the two sets, the summary and the summarized. In this article we use the cover relation defined over references since both queries and logs can be seen as sets of references, and thus the quality measure can be used on queries or on logs, or on any sets of references. Note that the definition of the measure is even more general in the sense that it does not rely on a particular cover relation. We now formalize these notions.

4.2 Definitions and Properties

We first introduce the notion of coverage of references.

Definition 8. (Coverage) A reference $r = \langle m_1, \ldots, m_n \rangle$ covers another reference $r' = \langle m'_1, \ldots, m'_n \rangle$ if $\forall_i \in [1, n]$, $m_i >_i m'_i$ or $m_i = m'_i$. For a set R of references, cover(R)=$\{f \in \Pi_{L_1^1}(D_1) \times \Pi_{L_1^2}(D_2) \times \ldots \times \Pi_{L_1^n}(D_n) \mid \exists r \in R, r\ covers\ f\}$.

Figure 2 illustrates this principle. We note L the set of references of some queries to be summarized, S the set of references of the summary, $K = L \cap S$, $D = L \setminus K$ and $A = S \setminus K$. The coverages of D and A are references (denoted by $cover(A)$ and $cover(D)$ and depicted with the same color as A and D respectively) in the most detailed level.

For instance, consider Example 5. $L = q_1 \cup q_2$, $S = lca(q_4)$, $K = \emptyset$ and $cover(A) = \{Pepsi, Coke, Orangina\} \times \{July08, August08, September08, October08, November08, December08\} \times \{Paris, Blois, Marseille\}$ with $|cover(A)| = 54$. $cover(D) = \{Pepsi, Coke\} \times \{July08\} \times \{Paris, Marseille\} \cup \{Coke\} \times \{July08\} \times \{Blois\}$ and $|cover(D)| = 5$. We have $cover(D) \subset cover(A)$. Intuitively, we expect a maximum recall and a bad precision because all covered references are recalled but a lot of other references are introduced.

To formalize this intuition, our measure of recall is the proportion of covered references existing in $cover(D)$ and found in $cover(A)$ compared with the set of references in $cover(D)$. Moreover, recall favours maximality of K. Our measure of precision is the proportion of covered references existing in $cover(D)$ and found in $cover(A)$ compared with the set of references in $cover(A)$. As for recall, precision encourages maximality of K. Of course if the summary is empty then the measure should be zero.

Definition 9. (hf-*measure*) Let L and R be two sets and $K = L \cap R$, $D = L \setminus K$ and $A = R \setminus K$. Let $\{D_1, \ldots, D_n\}$ be the set of dimensions defining the coverage. *h-recall*

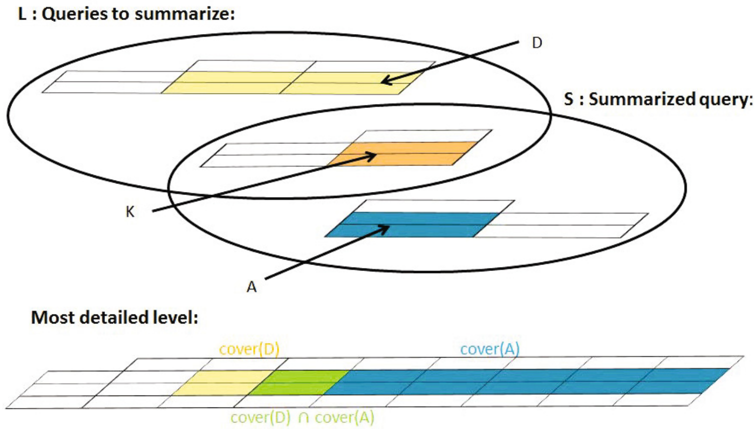


Fig. 2 Principle of the Quality Measure

is $r = \frac{|K \cup (cover(D) \cap cover(A))|}{|K \cup cover(D)|}$ and *h-precision* is $p = \frac{|K \cup (cover(D) \cap cover(A))|}{|K \cup cover(A)|}$. These measures are aggregated with the classical F-measure: $hf\text{-}measure\ (L, R, \{D_1, \ldots, D_n\}) = 2 \times \frac{p \times r}{p + r}$.

We conclude this section by noting that all operators of *QSL* maximize either *h-recall* or *h-precision*. Indeed, $\cup_B$ and *lca* lead to a *h-recall* of 1 and precision in 0 and 1, and all other operators lead to a *h-precision* of 1 and a recall in 0 and 1. Table 1 gives the range of values for *h-recall*, *h-precision*, recall and precision of each operator of *QSL*. The following property can easily be shown.

Property 1. Let L and R be two sets and $\{D_1, \ldots, D_n\}$ be a set of dimensions defining a coverage. $hf\text{-}measure\ (L, R, \{D_1, \ldots, D_n\}) = 1$ if and only if R and L cover exactly the same set of references.

Table 1 Table of *h-recall*, *h-precision*, recall and precision for each operator

operators	*h-precision*	*h-recall*	precision	recall
$\cup_B$	[0..1]	1	[0..1]	1
$\cap_B$	1	[0..1]	1	[0..1]
$\setminus_B$	1	[0..1]	1	[0..1]
$\lhd$	1	[0..1]	1	[0..1]
lca	[0..1]	1	[0..1]	[0..1]
max	1	[0..1]	1	[0..1]
mostfreq	1	[0..1]	1	[0..1]

5 Automatic Summarization of a Query Log

In this section, we present an algorithm for summarizing a log, based on *QSL* and our quality measure *hf-measure*. The main idea is that a summary of a log is also a log. We also present the properties of the summaries constructed with the algorithm.

5.1 SummarizeLog Algorithm

SummarizeLog algorithm is a greedy algorithm successively summarizing the queries of a log using *QSL* operators until a given length α for the summary is reached. The *QSL* expression used is that maximizing *hf-measure* while changing the log. Two strategies are defined for the choice of the expression. The first one checks for each query or each pair of consecutive queries what is the *QSL* operation maximizing *hf-measure*. The chosen expression is this particular operation (strategy 1). The second strategy checks for each pair of consecutive queries what is the *QSL* binary operation maximizing *hf-measure* and applies this operation. Then the strategy looks for on this result, the unary operation maximizing *hf-measure* (strategy 2). In this case, the *QSL* expression used is of the form $u(q\ b\ q')$ where u is a unary operator and b is a binary operator. In what follows, if q is a query resulting of a *QSL* expression, we call $queries(q)$ the set of queries involved in the *QSL* expression defining q.

Algorithm 5. SummarizeLog (strategy 1)

```
INPUT:
    $L$: A log
    $\mathcal{U}$: A set of unary operators
    $\mathcal{B}$: A set of binary operators
    $D$: A set of dimensions
    $\alpha$: A positive integer
OUTPUT: A summary of L
VARIABLES:
    $q_u, q_b$: Queries
    $max_u, max_b$: Real
while $|L| > \alpha$ do
    $max_u \leftarrow 0$
    for each $op \in \mathcal{U}$ do
        for each $q \in L$ do
            if $op(q) \neq q$ and $hf\text{-}measure(q, op(q), D) > max$ then
                $max_u \leftarrow hf\text{-}measure(q, op(q), D)$
                $q_u \leftarrow op(q)$
            end if
        end for
    end for
    $q_b \leftarrow argmax_2(\{hf\text{-}measure(q \cup q', op(q,q'), D) | op \in \mathcal{B}, L^{-1}(q) = L^{-1}(q') - 1\})$
    $max_b \leftarrow max(\{hf\text{-}measure(q \cup q', op(q,q'), D) | op \in \mathcal{B}, L^{-1}(q) = L^{-1}(q') - 1\})$
    if $max_u > max_b$ then
        replace in $L$ $queries(q_u)$ by $q_u$
    else
        replace in $L$ $queries(q_b)$ by $q_b$
    end if
end while
return $L$
```

Algorithm 6. SummarizeLog (strategy 2)

```
INPUT:
    $L$: A log
    $\mathcal{U}$: A set of unary operators
    $\mathcal{B}$: A set of binary operators
    $D$: A set of dimensions
    $\alpha$: A positive integer
OUTPUT: A summary of L
VARIABLES: $q_u, q_b$: Queries
while $|L| > \alpha$ do
    $q_b \leftarrow argmax_2(\{hf\text{-}measure(q' \cup q'', q''', D) | q''' = op(q', q''), op \in \mathcal{B}, q', q'' \in L\})$
    $q_u \leftarrow argmax_2(\{hf\text{-}measure(q_b, q'', D) | q'' = op(q_b), op \in \mathcal{U}\}$
    replace in $L$ $queries(q)$ by $q_u$
end while
return $L$
```

Let us illustrate briefly how strategy 1 operates on the toy example. Suppose it is called with the following parameters: $L = \langle q_1, q_2, q_3 \rangle$, $\mathcal{U}$ and $\mathcal{B}$ are respectively the sets of unary and binary operators of *QSL*, D is the set of dimensions of the toy example and $\alpha = 2$. All unary operators are applied on each query q_1, q_2, q_3 of the log and only the output that effectively summarizes the query is considered, i.e., a summary different from the query it summarizes and that achieves the best hf-*measure* (line 2–10). In our example, this is $lca(q_3)$. Then all binary operators are applied on each pair of consecutive queries q_1, q_2 and q_2, q_3. Again, only the summary achieving the best hf-*measure* is considered (line 11–12), in our example this is $q_1 \cup_B q_2$. Finally, among the two summaries considered, the one achieving the best hf-*measure* is used to produce the summary of the log at this step. In our example, the resulting summary at this step is $\langle q_1 \cup_B q_2, q_3 \rangle$. The algorithm then stops since the desired length of the summary, 2, is reached.

5.2 Properties of the Summaries

We first note that by construction, the summary S of a log L defines a partition of the log. Indeed, each query of S is defined by a *QSL* expression that involves a distinct subsequence of queries in L.

Property 2. (Partitioning) A summary $S = \langle s_1, \ldots, s_m \rangle$ of a log L defines a partition of L where each s_i summarizes with a *QSL* expression a non empty subsequence of L, the summarized sequences being pairwise disjoint and covering exactly L.

Using the properties of the *QSL* operators, we identify two sublanguages called respectively QSL^r and QSL^p. QSL^r is the language composed of operators maximizing the *h-recall* i.e., $QSL^r = \{\cup_B, lca\}$ and QSL^p is the language composed of operators maximizing *h-precision*, i.e., $QSL^p = \{\cap_B, \backslash_B, \lhd, mostfreq, max\}$. These two languages lead to the following simple properties. In what follows, we call for a query q, $member(q)$ the set of members appearing in q, i.e., $member(q) = \bigcup_i m_i(q)$ and for a set X of queries, $member(X) = \bigcup_{q \in X} member(q)$.

Property 3. (Query defined with QSL^r) Let q^r be a query defined with a QSL^r expression and let m be a member. If there is no member $m' \in member(q^r)$ such that $m' \geq m$ then $m \notin member(queries(q^r))$ and $\nexists m'' \in member(queries(q^r))$ such that $m > m''$. If $\exists m' \in member(q^r)$ such that $m > m'$ then $\exists m'' \in member(queries(q^r))$ such that $m > m''$.

This property states that if a summary is constructed only with operators maximizing *h-recall*, then every member not covered by a member appearing in the summary cannot appear in the queries involved in the expression. A dual property holds for *h-precision*.

Property 4. (Query defined with QSL^p) Let q^p a query defined with a QSL^p expression and m a member. If $m \in member(q^p)$ then $m \in member(queries(q^p))$.

These two properties extend straightforwardly to summaries.

Property 5. (Summary defined with QSL^r) Let S^r be a summary constructed with QSL^r expressions from a log L. If a member m is not covered by a member appearing in S^r, then neither m nor none m' covered by m can appear in L. If m covers some members of S^r, then m covers members of L.

Property 6. (Summary defined with QSL^p) Let S^p be a summary constructed with QSL^p expressions from a log L. A member m appearing in S^p appears necessarily in L.

The following section illustrates the interest of these properties.

6 Querying the Log Efficiently

In this section, we propose a language for searching a log. We first begin by describing how the properties given in the previous section allow for efficient searches in the log.

6.1 Using Summaries for an Efficient Search

If a query log is very large, and does not fit in main memory, searching for a member in this log can be very costly. We now describe how the basic properties of *QSL* operators can be used for efficient querying. Suppose that for a given log L, two summaries are available, the first one S^r constructed with QSL^r and the second one

S^p constructed with QSL^p. Consider a first boolean function called $lookup(m)$ that returns true if a member m is present in some queries of the log, or false otherwise. The *lookup* algorithm (see Algorithm 7), uses properties 2 to 6 to avoid accessing all the log.

Algorithm 7. lookup

```
INPUT:
    $L$: a log,
    $S^r$: a summary of $L$ constructed with $QSL^r$,
    $S^p$: a summary of $L$ constructed with $QSL^p$,
    $m$: a member.
OUTPUT: A boolean.
1:  if $m \in member(S^p)$ then
2:      return True
3:  end if
4:  if $\exists q \in queries(S^r)$ with $q = q_1 \cup_B \ldots \cup_B q_x$ and
    $m \in member(q)$ then
5:      return True
6:  end if
7:  for each $q \in queries(S^r)$ such that
    $\exists m' \in member(q)$ with $m' \geq m$ do
8:      for each $q' \in candidateQueries(q,m)$ do
9:          if $m \in q'$ then
10:             return True
11:         end if
12:     end for
13: end for
14: return False
```

Algorithm 8. candidateQueries

```
INPUT:
    $q$: a query,
    $m$: a member.
OUTPUT: A set of queries where $m$ may appear.
VARIABLE: A set of queries $Q$, a set of members $M$.
1:  $Q \leftarrow \emptyset$
2:  let $lca(e_1) \cup_B \ldots \cup_B lca(e_x) \cup_B q_1 \cup_B \ldots \cup_B q_y$ be the $QSL$
    expression defining $q$
3:  $M \leftarrow \{m' \in member(q) | m' \geq m\}$
4:  if $m \in M$ then
5:      $Q \leftarrow Q \cup \{q_1, \ldots, q_y\}$
6:  end if
7:  for each $m' \in M$ do
8:      for each $q'$ appearing in $lca(e_1) \cup_B \ldots \cup_B lca(e_x)$ do
9:          if ($q'$ appears in a number of compositions of lca
            $\leq level(m') - level(m)$) OR $m$ is DefaultMember
            then
10:             $Q \leftarrow Q \cup \{q'\}$
11:         end if
12:     end for
13: end for
14: return $Q$
```

Example 7. Consider the log of Example 1 and its summaries $S^r = \langle q'_1, q'_2 \rangle$ and $S^p = \langle q'_3 \rangle$, where $q'_1 = lca(q_1) = \langle \{Soda\}, \{S2\text{-}08\}, \{France\} \rangle$, $q'_2 = q_2 \cup_B q_3 = \langle \{Coke, Orangina\}, \{July08, S2\text{-}08\}, \{North, South\} \rangle$, and $q'_3 = q_1 \lhd q_2 \lhd q_3 = q_1 = \langle \{Pepsi, Coke\}, \{July08\}, \{Paris, Marseille\} \rangle$. The call to $lookup(Pepsi)$ requires only to access S^p to answer $true$ and the call to $lookup(2008)$ requires only to access S^p and S^r to answer $false$. $lookup(Orangina)$ requires only to access S^p and S^r to answer $true$ (cf. lines 4 to 6). To output $false$, $lookup(August08)$ requires to access S^p, S^r and finally q_1, but avoids the access to q_2 and q_3 since $August08$ cannot appear in the operands of an union whose result does not contain it (cf. lines 3 to 6 of $candidateQueries$).

lookup algorithm also serves as the basis for the algorithm $lookupCover(m)$, that particularly uses property 5. *lookupCover* returns true if there is at least one member covered by m in the log L and false otherwise.

Example 8. Consider the same queries of Example 7. The call to $lookupCover(2008)$ requires only to access S^p to answer $true$.

AlgorithmCover 9. lookupCover

```
INPUT:
    $L$: a log,
    $S^r$: a summary of $L$ constructed with $QSL^r$,
    $S^p$: a summary of $L$ constructed with $QSL^p$,
    $m$: a member.
OUTPUT: A boolean.
if $\exists m' \in member(S^p)$ with $m \geq m'$ then
    return True
end if
if $\exists q \in queries(S^r)$ and $\exists m' \in member(q)$ with $m \geq m'$ then
    return True
end if
for each $q \in queries(S^r)$ such that $\exists m' \in member(q)$ with $m' \geq m$ do
    for each $q' \in candidateCoveredQueries(q,m)$ do
        if $\exists m'' \in member(q')$ with $m \geq m''$ then
            return True
        end if
    end for
end for
return False
```

Algorithm 10. candidateCoveredQueries

```
INPUT:
    $q$: a query,
    $m$: a member.
OUTPUT: A set of queries where $m$ may appear.
VARIABLE: A set of queries $Q$, a set of members $M$.
$Q \leftarrow \emptyset$
let $lca(e_1) \cup_B \ldots \cup_B lca(e_x) \cup_B q_1 \cup_B \ldots \cup_B q_y$ be the $QSL$ expression defining $q$
$M \leftarrow \{m' \in member(q) | m' \geq m\}$
if $m \in M$ then
    $Q \leftarrow Q \cup \{q_1, \ldots, q_y\}$
end if
for each $q'$ appearing in $lca(e_1) \cup_B \ldots \cup_B lca(e_x)$ do
    $Q \leftarrow Q \cup \{q'\}$
end for
return $Q$
```

We introduce now function *getQueries*, returning the queries of the log where member m is present. It can be easily deduced from *lookup* by removing the first lines and outputting the relevant queries instead of a boolean. *getQueries* can also be used to find the queries where $m \times m'$ appears, since this corresponds to $getQueries(m) \cap getQueries(m')$, and thus it can also be used to query the log using references. *getQueries* is at the core of *getCoveredQueries* since it only requires to implement fully property 5.

Algorithm 11. getQueries

```
INPUT:
    $L$: a log,
    $S^r$: a summary of $L$ constructed with $QSL^r$,
    $m$: a member.
OUTPUT: A set of queries from $L$.
VARIABLES: A set of queries $Q$.
$Q \leftarrow \emptyset$
for each $q \in queries(S^r)$ such that $\exists m' \in member(q)$ with $m' \geq m$ do
    for each $q' \in candidateQueries(q,m)$ do
        if $m \in member(q')$ then
            $Q \leftarrow Q \cup \{q'\}$
        end if
    end for
end for
return $Q$
```

AlgorithmGet 12. getCoveredQueries

```
INPUT:
    $L$: a log,
    $m$: a member.
OUTPUT: A set of queries from $L$.
VARIABLES: A set of queries $Q$.
$Q \leftarrow \emptyset$
for each $q \in queries(L)$ do
    for each $q' \in candidateCoveredQueries(q,m)$ do
        if $\exists m' \in member(q')$ with $m \geq m'$ then
            $Q \leftarrow Q \cup \{q'\}$
        end if
    end for
end for
return $Q$
```

6.2 Querying a Log

In the previous subsection, we propose algorithms to search efficiently a member in the log. We now describe a language that enables to declaratively express complex searches for retrieving queries in a log. Consider the following simple queries on a log:

- Are there queries in the log that contain the members of the query q?
- Are there queries in the log that contain members covered by the members of q?
- What are the queries in the log that contain the members of q? That contain members covered by the members of q?
- What are the queries of the log that follow a query containing members the members of q?

To define operators for searching the log with a query expression as parameter, we define the two following relations over query expressions.

Definition 10. (Specialization relation over query expression) Let q and q' be two query expressions. q specialises q', noted $q \prec q'$, if $\forall i \in [1,n]$ and for all members $m' \in m_i(q')$, there is a member $m \in m_i(q)$ such that m' covers m.

Definition 11. (Inclusion of query expressions) Let q and q' be two query expressions. $q \sqsubseteq q'$ if for all i ∈ [1,n], $m_i(q) \subseteq m_i(q')$

Example 9. $\langle\{Soda\},\{all\},\{all\}\rangle$ is more general than $\langle\{Pepsi,Drink\},\{All\},\{All\}\rangle$. The opposite is not true. $\langle\{Pepsi\},\emptyset,\emptyset\rangle$ is included in $\langle\{Pepsi,Poke\},\{2008\},\{All\}\rangle$.

The search language is composed of two operators for querying a log. The first one is unary and allows to filter the log with a query. It is noted $filterLog(L,q,comp)$ where L is a log, q is a query expression and $comp$ is a comparison symbol, either $\sqsubseteq$ or $\prec$. The second operator is binary and allows to find neighbors of queries. It is noted $getNeighbor(L,L',dir)$ where L,L' are logs and dir is one of $succ, pred$. These two operators output a log of queries as answer. We now give the formal definitions.

Definition 12. Let L be a log, q a query expression and $comp$ a comparator in $\{\sqsubseteq,\prec\}$, $filterLog(L,q,comp) = L'$ where L' is the restriction of L to the set $\{a_1,\ldots,a_p\}$ such that for all a_i, q $comp$ $L(a_i)$ is true and for all $x \in \{1,n\} \setminus \{a_1,\ldots,a_p\}$, q $comp$ $L(x)$ is false. Let L be a log, L' be a log such that $pos(L') \subset pos(L)$, with $pos(L') = \{a_1,\ldots,a_p\}$, $getNeighbor(L,L',dir) = L'' \subset L$ where, if dir is $succ$ (resp. $pred$), $pos(L'') = \{a_1+1,\ldots,a_p+1\}$ (resp. $pos(L'') = \{a_1-1,\ldots,a_p-1\}$) and for all p in $pos(L'')$, $L''(p) = L(p)$ if defined.

$filterLog(L,q,comp)$ can be implemented naively by scanning L. A more efficient implementation is proposed in Algorithm 13, where *candidateQueries* (resp., *candidateCoveredQueries*) is used for accessing only the relevant parts of the log L when $comp$ is $\sqsubseteq$ (resp., $\prec$). We illustrate these operators with some simple searches over the running example.

Example 10. Let us query the log $L = \langle q_1, q_2, q_3 \rangle$ where q_1, q_2, q_3 are the query expressions of Example 1. The query: "is member Perrier in the log?" is expressed by: $filterLog(L, \langle \{Perrier\}, \emptyset, \emptyset \rangle, \sqsubseteq)$ As this expression returns the empty set, the answer is interpreted as no. The query "what are the queries covered by Pepsi and S2–08?" is expressed by $filterLog(L, \langle \{Pepsi\}, \{S2\text{-}08\}, \{All\} \rangle, \prec$ $)$ which returns $\langle q_1 \rangle$. The query "what are the queries that immediately follow those queries covered by Pepsi and S2–08" is expressed by $getNeighbor(L, filterLog(L, \langle \{Pepsi\}, \{S2\text{-}08\}, \{All\} \rangle, \prec), succ)$ and returns $\langle q_2 \rangle$. Finally note that summaries can also be used to query logs. Indeed SummarizeLog can be seen as an operator that outputs a log by summarizing another log. For instance, the expression $L' = SummarizeLog(filterLog(L, \langle \emptyset, \emptyset, \{North\} \rangle, \sqsubseteq))$ summarizes only queries q_2 and q_3 and $filterLog(L', \langle \{Drink\}, \emptyset, \emptyset \rangle, \sqsubseteq)$ checks if member Drink is used to summarize those queries.

Algorithm 13. filterLog

```
INPUT:
    $L$: A log,
    $q$: A query expression
    $comp$: A comparator
OUTPUT: A log.
VARIABLES: A set of queries $C$.
Let $S^r$ be a summary of $L$ constructed with $QSL^r$
$C \leftarrow \emptyset$
for each $m_i(q)$ do
    for each $m \in mi(q)$ do
        if $comp = \sqsubseteq$ then
            for each $q' \in queries(S^r)$ such that $\exists m' \in member(q')$ with $m' \geq m$ do
                $C \leftarrow C \cap candidateQueries(q, m)$
            end for
        else
            $C \leftarrow C \cap candidateCoveredQueries(q, m)$
        end if
    end for
end for
return $L|_{\{L^{-1}(q') | q' \in C, q\ comp\ q'\}}$ {Access to $L$}
```

6.3 Use Case: Defining New Analytical Sessions

We conclude the section by presenting a realistic use case recapitulating the interest of summarizing and querying a log of OLAP queries.

Let L be a log containing a large number of past queries focused on the sales of various products.

We suppose that a user wishes to conduct a new analysis. In order to prepare his analysis, he decides to visualize a summary of L composed with only ten queries.

For summarizing a log by generalizing it, the *SummarizeLog* operator will use the QSL^r language with *Strategy* 2 because the *lca* operator (generalizing the queries) is used in each step of summarization.

Thus, the user applied the function $SummarizeLog(L)$ that outputs a summary of L.

We suppose that the user decides to conduct a new analysis about the *cola sodas*. Visualizing the summary, he notes no queries of the summary are composed with *Coke* products. However, *Soda* appears in these queries. Because *Soda* generalizes *Coke*, the user has to check if queries of the initial log are involved in an analysis about *Coke*. Therefore, he filters the initial log by using the function $filterLog(L, \langle\{Coke\}, \emptyset, \emptyset\rangle, \sqsubseteq)$. A sequence of queries L_{filter} is returned to the user. Thus, he follows these query examples for forming the first query of his new analysis session. For the rest of his analysis session, the user decides to obtain the queries immediately following the queries of L_{filter} by using the function $getNeighbor(L, filterLog(L, \langle\{Coke\}, \emptyset, \emptyset\rangle, \sqsubseteq), succ)$.

7 Implementation and Tests

The framework is implemented with Java 6. The implementation has been done considering that dimensions fit in main memory. These dimensions are represented by trees storing for each member the cardinality of its coverage at the most detailed level. Tests have been run on a computer equipped with Intel Core 2 Duo CPU E8400 clocked at 3.00 GHz with 3.48 Go of usable RAM, under windows 7 ultimate edition. The logs used are synthetic logs on the Foodmart database example coming with the Mondrian OLAP engine. The process of log generation is detailed in [Giacometti et al., 2011] and aims at simulating real sessions. This process is based on a random choice between the DIFF and RELAX operators (described in [Sarawagi, 1999] and [Sarawagi, 2000]), applied on the data of the Foodmart database. These operators can automatically explore a cube by a sequence of drill-downs or roll-ups, identifying interesting differences between cell pairs. We suppose that these differences are likely to be identified by a real user, hence the simulation of an OLAP analysis.

The query generator is parametrized by a number of dimensions, called the *density*, that represents the number of dimensions used for navigation. Another parameter indicates the maximum number of queries per session. For our tests, we have used logs of high density (5 dimensions out of the 13 available are used to simulate the navigation) and low density (13 dimensions are used to simulate the navigation). The high density logs are respectively composed of 119, 242, 437 and 905 queries. The low density logs are respectively composed of 121, 239, 470 and 907 queries. In what follows, the length of summaries are expressed as a ratio of the original log size.

We have conducted a large set of tests, and we report the main results here in 4 categories:

- Study of the quality measure,
- Assessment of the two strategies proposed for SummarizeLog,
- Sensitivity of the approach to the log density,
- Efficiency of the operators for searching logs.

7.1 Study of the Quality Measure

The aim of our first tests is to study our quality measure. We begin by assessing the overall usage of each operator of *QSL* existing in the *QSL* expression built by SummarizeLog. Figure 3 shows for *QSL* that the hf-measure favors the union and lca operator, and that the $\setminus_B$ operator is never used.

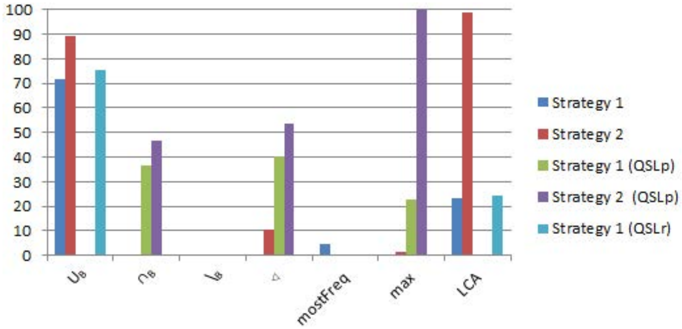


Fig. 3 Usage of the operators of *QSL*, QSL^p and QSL^r in two strategies on logs of high density

We note *hf-measure* favours the *lca* and $\cup_B$ operators. This demonstrates that the sublanguage QSL^r, which is used for implementing the search operators efficiently, is indeed of particular interest.

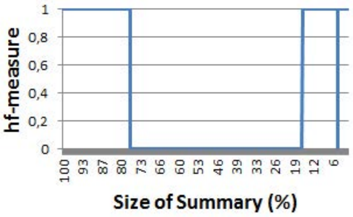


Fig. 4 Overall quality for *QSL* on 905 queries of high density in strategy 1

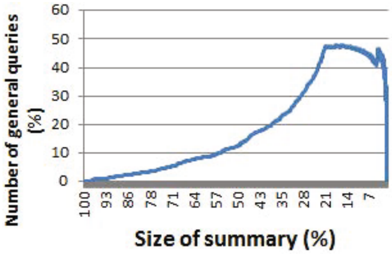


Fig. 5 Ratio of general queries for *QSL* on 905 queries of high density in strategy 1

Our second test is to compute the ratio of general queries a summary contains. A general query is a query having only the All member in each dimension. Such queries reveal little information to the user and thus their appearance in summaries should be limited. Figures 5, 10, 11, 15, 16 show that the ratio of general queries increases as the length of the summary decreases, as expected. We note that the number of general queries never exceeds 50 % of the number of queries in the summary. This test shows that our quality measure can limit these queries and favours more interesting ones.

Finally we investigate the usefulness of the quality measure to assess the overall quality of the summaries. This overall quality is evaluated as follows: For each query q of the summary, we evaluate its quality using *hf-measure* between its references and the references of the queries of the log that q summarizes. The overall quality of the summary is the minimum, i.e., the worst, of the qualities of all queries of the summary. Interestingly, Figures 4 shows that this overall quality is eventually good for summaries of small length. It can also be seen on Figures 8, 9, 17 and 18, for logs of different lengths.

7.2 Assessment of the Two Strategies

This series of tests assess the efficiency and effectiveness of the two strategies proposed for SummarizeLog. The behaviours reported below are observed whatever the log length. Figures 6 (with a logarithmic scale) and 7 report the computation time needed for summarizing.

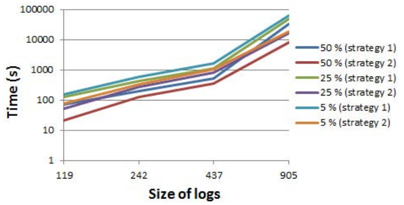


Fig. 6 Efficiency for *QSL* for the two strategies on logs of high density

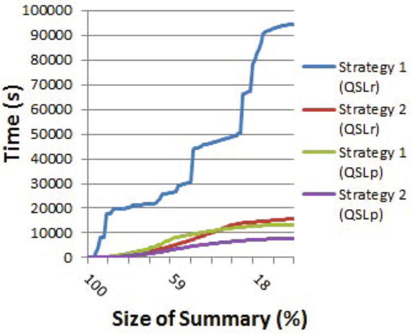


Fig. 7 Efficiency for QSL^p and QSL^r for the two strategies on 905 queries of high density

Note that, as expected, the computation time is polynomial in the length of the log. This time is quite expensive for large logs. Strategy 2 is globally more efficient, requiring less quality tests (the most expensive part of SummarizeLog). Note that due to the fact that *hf-measure* is evaluated on references and computes a coverage at the lowest level of details, the computation time for languages including $\cup_B$ can be extremely high. Indeed, for QSL^r, strategy 1 can result in successive unions that produce queries that are large sets of references, whereas strategy 2 systematically uses *lca* that reduces the number of references. Figures 8 and 9 show the overall quality of the summaries produced with the two strategies. It can be seen that strategy 1 globally achieves a better quality than strategy 2. This can be explained by the fact that strategy 1 explores more combinations of *QSL* operators than strategy 2.

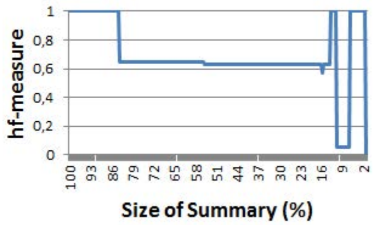


Fig. 8 Overall quality for *QSL* on 242 queries of high density for strategy 1

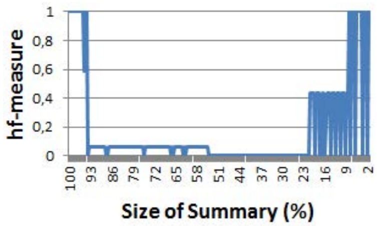


Fig. 9 Overall quality for *QSL* on 242 queries of high density for strategy 2

Finally, Figures 10, 11 and 12 indicate the ratio of general queries in the summaries constructed with each of the strategies.

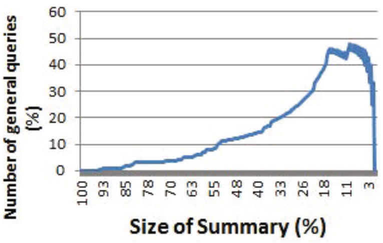


Fig. 10 Ratio of general queries for *QSL* on 242 queries of high density for strategy 1

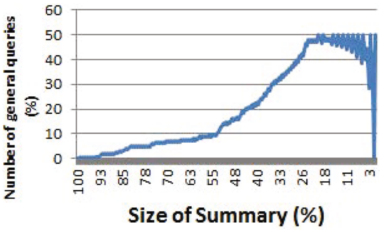


Fig. 11 Ratio of general queries for *QSL* on 242 queries of high density for strategy 2

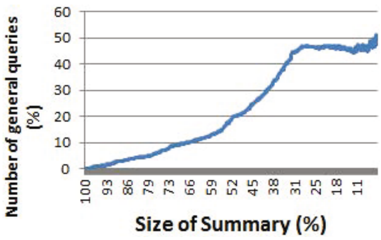


Fig. 12 Ratio of general queries for *QSL* on 905 queries of high density for strategy 2

It can be seen that strategy 1 and 2 have a similar behaviour, the number of general in queries produced with strategy 1 increasing more slowly. Consequently, strategy 1 produces fewer general queries than strategy 2 which confirms that strategy 1 computes summaries of better quality.

7.3 Sensitivity to the Log Density

Figures 13, 14, 15, 16, 17, 18 report the result of tests on logs of similar lengths but different densities, in terms of efficiency, global quality and ratio of general queries. It can be seen that SummarizeLog achieves both a better quality and a better computation time on logs of high density, as expected. Remarkably, even on logs of low density, the ratio of general queries remains acceptable.

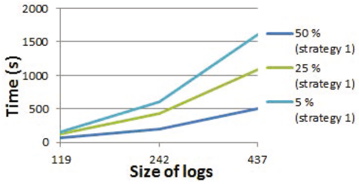


Fig. 13 Efficiency for *QSL* on three logs of high density for strategy 1

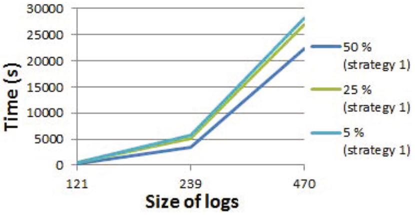


Fig. 14 Efficiency for *QSL* on three logs of low density for strategy 1

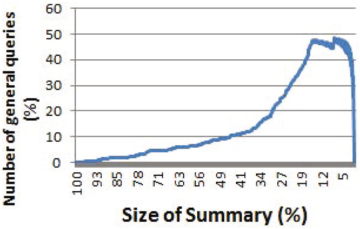


Fig. 15 Ratio of general queries for *QSL* on 437 queries of high density for strategy 1

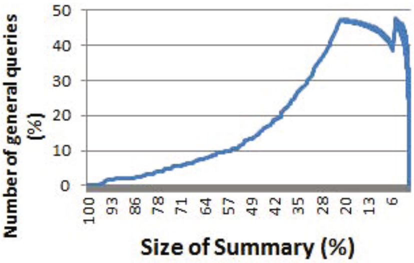


Fig. 16 Ratio of general queries for *QSL* on 470 queries of low density for strategy 1

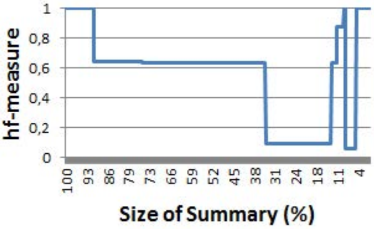


Fig. 17 Overall quality for *QSL* on 437 queries of high density for strategy 1

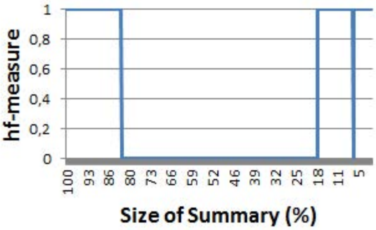


Fig. 18 Overall quality for *QSL* on 470 queries of low density for strategy 1

7.4 *Efficiency of the Search Operators*

Our last series of tests assess the efficiency of the Lookup algorithm, that is at the core of the search operators. This operator relies on two summaries constructed respectively with QSL^p and QSL^r. Figures 19 and 20 show the proportion of general queries produced by QSL^r (Note that summaries constructed with QSL^p cannot have general queries unless already present in the initial log, which for our tests was not the case.).

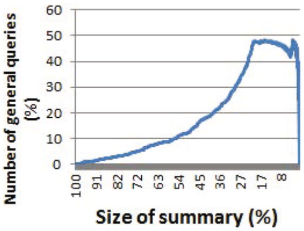


Fig. 19 Ratio of general queries for QSL^r on 905 queries of high density for strategy 1

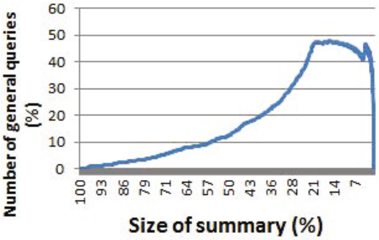


Fig. 20 Ratio of general queries for QSL^r on 907 queries of low density for strategy 1

Figures 21 and 22 show the average gain in efficiency for a lookup search of 3210 members chosen randomly, from two summaries computed with QSL^p and QSL^r for strategy 1 on logs of high and low density. The gain is the ratio of computation time between the lookup algorithm and a basic scan with disk accesses of the log file.

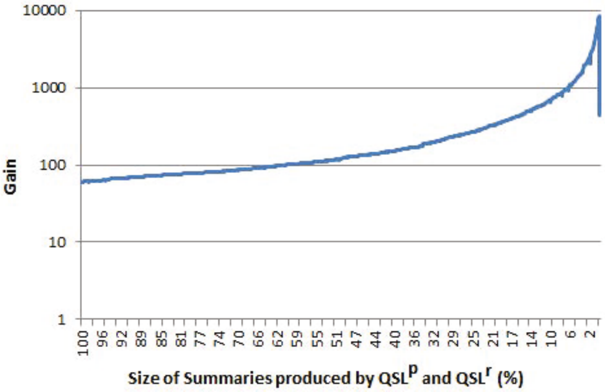


Fig. 21 Gain for Lookup Algorithm on 905 queries of high density for strategy 1

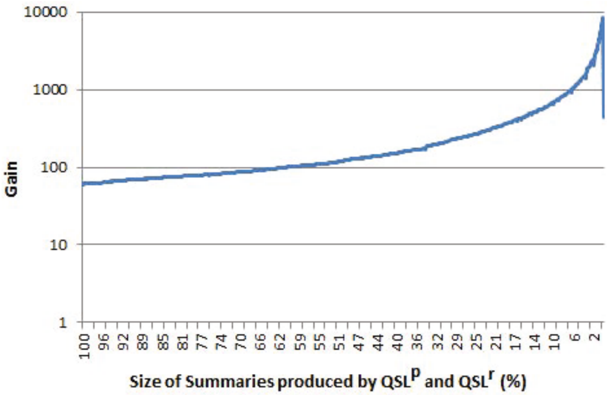


Fig. 22 Gain for Lookup Algorithm on 907 queries of low density for strategy 1

It can be seen that the gain is in favour of lookup algorithm whatever the density.

8 Related Work

Summarization of structured data has attracted a lot of attention in various domain, covering web server log [Zadrozny and Kacprzyk, 2007] pattern mining (see e.g., [Ndiaye et al., 2010] that includes a brief survey), sequences of event [Peng et al., 2007], database instance [Saint-Paul et al., 2005], multidimensional data stream [Pitarch et al., 2010], database workloads [Chaudhuri et al., 2003] and datacubes [Lakshmanan et al., 2002].

Many of these works rely on fuzzy set theory [Zadrozny and Kacprzyk, 2007, Saint-Paul et al., 2005] and/or are compression techniques for which it is important that original data can be regenerated [Lakshmanan et al., 2002, Ndiaye et al., 2010]. Moreover, it can be the case that the summary has not the same type as the data it summarizes. In the domain of databases [Saint-Paul et al., 2005, Pitarch et al., 2010, Lakshmanan et al., 2002], summarizing is applied to the database instance where, for OLAP data, measure values are taken into account.

In this paper we address the problem of summarizing an OLAP server query log. Our approach has the following salient features:

- We do not summarize a database instance, but a sequence of database queries.
- Summaries can be expressed declaratively with a manipulation language or constructed automatically.
- We do not assume any imprecise description of the members used in the queries, that could e.g., be described via fuzzy set theory. Instead, we only need the information at hand, i.e., the hierarchies described in the dimension tables.
- The type of the summary is the same as the type of the original data.
- We do not address the problem of regenerating the data from the summary, instead we focus on how to use summaries to efficiently search the log.

To the best of our knowledge, no work have yet addressed the problem of summarizing a database query log in a suitable and concise representation. As pointed out in [Khoussainova et al., 2009], many RDBMs provide query logging, but logs are used essentially for physical tuning, and noticeably, the term workload is often termed instead of log. Notable papers are [Chaudhuri et al., 2003] for relational databases and [Golfarelli, 2003] for multidimensional databases. [Chaudhuri et al., 2003] defines various primitives for summarizing query logs, essentially to filter it. The model of queries used covers both the query expression and query evaluation information (indexes used, execution cost, memory used, etc.) In this work, summarization aims at satisfying a given objective function for assisting DBA like finding queries in the log that have a low index usage. In [Golfarelli, 2003], logs are analysed for identifying views to materialize, using an operator that resembles our *lca* operator.

Usually, when a query log is displayed, often in flat table or file, it is not suitable for browsing or searching into it. In our earlier work [Colas et al., 2010], we propose to organize an OLAP query log under the form of a website. But if the log is large, browsing this website may be tedious. An effective log visualization and browsing tool is yet to be designed, and the present work is a step in that direction.

9 Conclusion and Perspectives

In this article, we propose a framework for summarizing and querying OLAP query logs. This framework relies on the idea that a query can summarize another query and that a log can summarize another log. Our contributions include a query manipulation language that allows to declaratively specify a summary, and an algorithm for automatically computing a query log summary of good quality. We also propose operators for querying OLAP query logs and show how summaries can be used to achieve an efficient implementation. The framework has been implemented and tests were conducted to show its interest.

Future work include the development and study of the different languages proposed in this article, as well as the validation of the approach on real and large query logs. Our long term goal is to study how query logs can support effectively the On-Line Analysis Process. Our future work will thus include the extension of our framework to a collaborative context where a log, composed of many sequences of queries performed by different users, each with a particular goal in mind, can be efficiently browsed and searched. Another direction is the generalisation of our framework to other types of logs (like web query logs for instance).

References

[Aligon et al., 2010] Aligon, J., Marcel, P., Negre, E.: A framework for summarizing a log of OLAP queries. In: IEEE ICMWI, Special Track on OLAP and Data Warehousing (2010)

[Aligon et al., 2011] Aligon, J., Marcel, P., Negre, E.: Résumé et interrogation de logs de requêtes OLAP. In: Proc. 11ème Conférence Internationale Francophone sur l'Extraction et la Gestion des Connaissances EGC (2011)

[Chatzopoulou et al., 2009] Chatzopoulou, G., Eirinaki, M., Polyzotis, N.: Query Recommendations for Interactive Database Exploration. In: Winslett, M. (ed.) SSDBM 2009. LNCS, vol. 5566, pp. 3–18. Springer, Heidelberg (2009)

[Chaudhuri et al., 2003] Chaudhuri, S., Ganesan, P., Narasayya, V.R.: Primitives for Workload Summarization and Implications for SQL. In: VLDB, pp. 730–741 (2003)

[Colas et al., 2010] Colas, S., Marcel, P., Negre, E.: Organisation de log de requêtes OLAP sous forme de site web. In: EDA 2010. RNTI, vol. B-6, pp. 81–95, Cépaduès, Toulouse (2010)

[Garcia-Molina et al., 2008] Garcia-Molina, H., Ullman, J.D., Widom, J.: Database Systems: The Complete Book. Prentice Hall Press, Upper Saddle River (2008)

[Giacometti et al., 2009] Giacometti, A., Marcel, P., Negre, E.: Recommending Multidimensional Queries. In: Pedersen, T.B., Mohania, M.K., Tjoa, A.M. (eds.) DaWaK 2009. LNCS, vol. 5691, pp. 453–466. Springer, Heidelberg (2009)

[Giacometti et al., 2011] Giacometti, A., Marcel, P., Negre, E., Soulet, A.: Query Recommendations for OLAP Discovery-Driven Analysis. IJDWM 7(2), 1–25 (2011)

[Golfarelli, 2003] Golfarelli, M.: Handling Large Workloads by Profiling and Clustering. In: Kambayashi, Y., Mohania, M., Wöß, W. (eds.) DaWaK 2003. LNCS, vol. 2737, pp. 212–223. Springer, Heidelberg (2003)

[Khoussainova et al., 2009] Khoussainova, N., Balazinska, M., Gatterbauer, W., Kwon, Y., Suciu, D.: A case for a collaborative query management system. In: CIDR (2009)

[Khoussainova et al., 2011] Khoussainova, N., Kwon, Y., Liao, W.-T., Balazinska, M., Gatterbauer, W., Suciu, D.: Session-Based Browsing for More Effective Query Reuse. In: Bayard Cushing, J., French, J., Bowers, S. (eds.) SSDBM 2011. LNCS, vol. 6809, pp. 583–585. Springer, Heidelberg (2011)

[Lakshmanan et al., 2002] Lakshmanan, L.V.S., Pei, J., Han, J.: Quotient cube: How to summarize the semantics of a data cube. In: VLDB, pp. 778–789. Morgan Kaufmann (2002)

[Ndiaye et al., 2010] Ndiaye, M., Diop, C.T., Giacometti, A., Marcel, P., Soulet, A.: Cube Based Summaries of Large Association Rule Sets. In: Cao, L., Feng, Y., Zhong, J. (eds.) ADMA 2010, Part I. LNCS, vol. 6440, pp. 73–85. Springer, Heidelberg (2010)

[Peng et al., 2007] Peng, W., Perng, C., Li, T., Wang, H.: Event summarization for system management. In: Berkhin, P., Caruana, R., Wu, X. (eds.) KDD, pp. 1028–1032. ACM (2007)

[Pitarch et al., 2010] Pitarch, Y., Laurent, A., Poncelet, P.: Summarizing Multidimensional Data Streams: A Hierarchy-Graph-Based Approach. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010 Part II. LNCS, vol. 6119, pp. 335–342. Springer, Heidelberg (2010)

[Saint-Paul et al., 2005] Saint-Paul, R., Raschia, G., Mouaddib, N.: General purpose database summarization. In: VLDB, pp. 733–744 (2005)

[Sarawagi, 1999] Sarawagi, S.: Explaining differences in multidimensional aggregates. In: VLDB, pp. 42–53 (1999)

[Sarawagi, 2000] Sarawagi, S.: User-adaptive exploration of multidimensional data. In: VLDB, pp. 307–316 (2000)

[Stefanidis et al., 2009] Stefanidis, K., Drosou, M., Pitoura, E.: "You May Also Like" Results in Relational Databases. In: PersDB (2009)

[Zadrozny and Kacprzyk, 2007] Zadrozny, S., Kacprzyk, J.: Summarizing the contents of web server logs: A fuzzy linguistic approach. In: FUZZ-IEEE, pp. 1–6. IEEE (2007)

Part II
Ontology and Semantic

A Complete Life-Cycle for the Semantic Enrichment of Folksonomies

Freddy Limpens, Fabien Gandon, and Michel Buffa

Abstract. Tags freely provided by users of social tagging services are not explicitly semantically linked, and this significantly hinders the possibilities for browsing and exploring these data. On the other hand, folksonomies provide great opportunities to bootstrap the construction of thesauri. We propose an approach to semantic enrichment of folksonomies that integrates both automatic processing and user input, while formally supporting multiple points of view. We take into account the social structure of our target communities to integrate the folksonomy enrichment process into everyday tasks. Our system allows individual users to navigate more efficiently within folksonomies, and also to maintain their own structure of tags while benefiting from others contributions. Our approach brings also solutions to the bottleneck problem of knowledge acquisition by helping communities to build thesauri by integrating the manifold contributions of all their members, thus providing for a truly socio-semantic solution to folksonomy enrichment and thesauri construction.

1 Introduction

Social tagging is a successful means to involve users in the life cycle of the content they exchange, read or publish online. However, folksonomies resulting from this practice have shown limitations, in particular, the spelling variations of similar tags and the lack of semantic relationships between tags that significantly hinder the navigation within tagged corpora.

Freddy Limpens · Fabien Gandon
Edelweiss - INRIA Sophia Antipolis, France
e-mail: fdy@pl-area.net, fabien.gandon@inria.fr

Michel Buffa
I3S - CNRS / University of Nice - Sophia Antipolis, France
e-mail: buffa@unice.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 127–150.
DOI: 10.1007/978-3-642-35855-5_7

One way of tackling these limitations is to semantically structure folksonomies. This can help navigate within tagged corpora by (1) enriching tag-based search results with spelling variants and hyponyms, or (2) suggesting related tags to extend the search, or (3) semantically organizing tags to guide novice users in a given domain more efficiently than with flat lists of tags or occurrence-based tag clouds, or (4) assisting disambiguation.

We present our approach to design a tagging-based system that integrates collaborative and assisted semantic enrichment of the community's folksonomy. We propose formal models and methods to support diverging points of view regarding the semantics of tags and to efficiently combine them into a coherent and semantically structured folksonomy.

Our end-user is the Ademe agency[1] which seeks to broaden the audience of its scientific work in the field of sustainable development and environmental issues. In this scenario, we can distinguish three types of stakeholders: (1) the expert engineers working at Ademe who are specialists of a given domain, (2) the archivists who take care of the indexing of the documents from Ademe and have transversal knowledge of the thematic covered at the agency, and (3) the public audience who has access to the documents of Ademe from its website. The archivists seek to both enrich their indexing base, which can be seen as a controlled folksonomy, and to upgrade it towards a thesaurus-like structure. The difficulty here comes from the different points of view that may arise from the community of expert engineers, and possibly also from the public, and that have to be turned into a coherent structure by the archivists.

In section two we present current works in folksonomy semantic enrichment, and position our contribution. In section three we give a general presentation of our approach. In section four we present the results of automatic processing of tag data, and detail our method to extract emergent semantics with a combination of string edit distances. Section five will cover the capture and exploitation of users contribution to provide a semantically enriched folksonomy that supports multiple points of view. Section six will conclude and give some insights about possible future developments.

2 Related Work

Folksonomy enrichment has been addressed by numerous research works covering a broad variety of approaches.

2.1 Extracting the Emergent Semantics

A first category of work aims at extracting the emergent tag semantics from folksonomies by measuring the semantic similarity of tags. The studies from

[1] French Environment and Energy Management Agency, `http://www.ademe.fr`

[Markines et al., 2009] and [Cattuto et al., 2008] propose an analysis of the different types of similarity measures and the semantic relations they each tend to convey. The simplest approach consists in counting the co-occurrence of tags in different contexts (users or resources). Cattuto *et al.* [Cattuto et al., 2008] showed that this type of measure provided subsumption relations but was not sufficiently accurate. More elaborate methods exploit the network structure of folksonomies making use of the distributional hypothesis that states that words used in similar contexts tend to be semantically related. To apply this hypothesis on tags, [Cattuto et al., 2008] computed the cosine similarity measure in the vector spaces obtained by folding the tripartite structure of folksonomy onto distributional aggregations spanning the associations of tags with either: the other tags (tag-tag context), or the users (tag-user context), or the resources (tag-resources). Their study shows that the tag-tag context performed best at a reasonable cost and that the semantic relation conveyed by this measure was of type "related". Mika [Mika, 2005] also applied and evaluated different folding of the tripartite structure of folksonomies. Interestingly, he showed according to a qualitative evaluation that exploiting user-based associations of tags yielded more representative taxonomic relations. The principle of this association is that if, *e.g.* the community of users using the tag "biological agriculture" is included in the community of users of the tag "agriculture", then the tag "agriculture" is broader than the tag "biological agriculture". Heyman *et al.* [Heymann and Garcia-Molina, 2006] proposed an algorithm that constructs a taxonomy from tags by crawling the similarity graph computed from the cosine distance based on the Tag-Resource context. The hierarchy of tags is built starting from the tag with the highest centrality, and each tag, taken in order of centrality, is added either as a child of one of the nodes or of the root node depending on a threshold value.

2.2 Models and Tools to Structure Tags

Another type of approach consists in letting users semantically structure tags. [Tanasescu and Streibel, 2007] proposed to tag the tags, [Huynh-Kim Bang et al., 2008] proposed a simple syntax to specify subsumption (with ">" or "<") or synonymy (with "=") relations between tags. Some tools available online also feature semantic structuring capacities such as Gnizr[2] and Semanlink[3], and even Flickr with machine tags[4]. In the same trend, the Linked Data community seeks to weave together the content of social web sites thanks to a set of formal ontologies not aimed at describing the knowledge of the communities but rather the structure of their knowledge exchange platforms. For instance SCOT[5] describes tags as parts of sharable tag clouds, and SIOC[6] describes online

[2] http://code.google.com/p/gnizr/

[3] http://www.semanlink.net

[4] http://www.flickr.com/groups/mtags/

[5] http://scot-project.org/

[6] http://sioc-project.org/

communities' content. MOAT[Passant and Laublet, 2008] is an ontology aimed at linking each tagging action with a URI representing the meaning of this tag action. These URIs can link to formal ontologies concepts or any web page containing a description of a notion. Once tag actions are formally linked to concepts, it is possible to disambiguate tags when searching, but also to exploit inference mechanisms via the formal concepts and to get a richer browsing experience. NiceTag[7] is a model that seeks to account for the usages of tags through a finer modeling of the relations between tags and the tagged resources [Limpens et al., 2009]. Its flexibility and the use of a named graphs mechanism allows this model to serve as a pivot model for all other tag models, adding a level of pragmatics. Finally, as we propose to support diverging points of view, let us recall briefly some multi-points of view approaches such as [Ribière, 1999] who proposed multi-points of view knowledge representations grounded on the conceptual graphs formalisms in which the links between concepts can be bound to a given point of view. [Bouquet et al., 2004] does not exactly propose representing concepts according to multiple points of view, but instead suggest contextualizing ontologies thanks to C-OWL, an extension of OWL. The idea of C-OWL is to provide a set of primitives to describe mappings between a series of « local » ontologies that can be each associated to a point of view.

Some other works seek to integrate one or several of the preceding approaches. For instance [Angeletou et al., 2008] and [Specia and Motta, 2007] make use of similarity metrics to find related tags, and then map these tags to concepts from available online ontologies in order to semantically structure tags with formal properties. [Van Damme et al., 2007] proposed an integrated approach to folksonomy enrichment including as many resources as possible, using each in a tailored way in addition to the validation of the inferences by the users.

Finally, our approach can be related to ontology construction and ontology maturing. Indeed, our approach clearly echoes attempts to build formal ontologies from texts [Aussenac-Gilles et al., 2000] or databases maintained by communities of users [Golebiowska, 2002]. More recently, Braun *et al.* [Braun et al., 2007] addressed the problem of collaborative ontology editing and pointed out the limitations of current ontology engineering tools in that respect. They proposed integrating ontology maturing in common tasks such as information seeking, and they developed a bookmarking service with the possibility for all users to add or edit new "semantic" tags formally structured with SKOS[8].

2.3 Discussion of Current Approaches

Full automatization of semantically enriching folksonomies is difficult. First the similarity measures used by [Cattuto et al., 2008, Markines et al., 2009], [Specia and Motta, 2007] or other methods for retrieving taxonomical structures

[7] http://ns.inria.fr/nicetag/2009/09/25/voc

[8] http://www.w3.org/2004/02/skos/core#

from folksonomies [Mika, 2005, Heymann and Garcia-Molina, 2006] are useful to bootstrap the process, but their accuracy in reflecting the communities knowledge is limited. The semantic grounding of these measures proposed by [Cattuto et al., 2008] can also help evaluate their accuracy. However, as this evaluation requires that tags be present in Wordnet synsets or in other ontological resources, the validity of these measures can only be evaluated for common knowledge and not really for specific terms that consist in one of the most valuable benefits of folksonomies. The same argument can be used towards other approaches [Angeletou et al., 2008] that make use of ontological resources to formally structure folksonomies.

On the other hand, approaches that rely on user input (to tag the tags, or to link a tag to an unambiguous concept) may induce, without user-friendly interfaces tailored to usages, a cognitive overload that regular users of tagging are not ready to bear. Integrated approaches try to overcome this limit by mixing automatic handling with user validation. However, none of these two types of approaches formally takes into account the multiplicity of points of view within a community, a feature at the core of our approach for which we will now give an overview.

3 Semantic Enrichment of Folksonomies

A generic method to semantically enrich all types of folksonomies in a fully automatic manner seems out of reach today. Our approach to semantically enriching folksonomies consists in creating a synergistic combination of automatic handling, to bootstrap the process, and of users' contributions at the lowest possible cost through user-friendly interfaces. We propose a system that supports conflicting points of view regarding the semantic organization of tags, but also helps online communities build a consensual point of view emerging from individual contributions.

3.1 SRTag: Using Named Graphs to Keep Track of Diverging Points of View

In order to model the semantic structuring of folksonomies while supporting conflicting views, we propose an RDF schema, SRTag[9], which makes use of named graphs mechanisms[Carroll et al., 2005, Gandon et al., 2007]. Named graphs allow to reify the semantic relationship between two tags or two concepts (modeled with SKOS) without the burden of classical RDF reification[10] (see figure 1). The benefits and the reasons for using named graphs to capture assertional intents are given in details in [Limpens et al., 2009], but we can merely recall here that we required a mechanism that allow to encapsulate statements about tags and give a URI to these

[9] http://ns.inria.fr/srtag/2009/01/09/srtag.html

[10] http://www.w3.org/TR/rdf-mt/#Reif

statement in order to be able to link them to other entities. For example, we wanted to be able to say that "Kevin agrees with the fact that `soil pollution` is a more specific term than `pollution` but Alex disagrees". Using a named graph that encapsulate "`soil pollution` is a more specific term than `pollution`" allows us to reuse it with as many other agreement or disagreement relations (or any other type of relation if needed). In addition, these named graphs are typed with our class `srtag:TagSemanticStatement` or with more precise subclasses.

The relationships between tags can be taken from any model, but we chose to limit the number of possible relations to thesaurus-like relations as modeled in SKOS. Then we modeled a limited series of semantic actions which can be performed by users (represented using `sioc:User` class), namely `srtag:hasApproved`, `srtag:hasProposed`, and `srtag:hasRejected`. We are then able to capture and track back users opinions (reject or approve) on the asserted relations, and thus to collect diverging points of view.

We distinguish different types of automatic and human agents according to their role in the life cycle of the folksonomy. We modeled different subclasses of the class `sioc:User` in order to filter statements according to the users who approve it. This includes `srtag:SingleUser` which corresponds to regular users of the system, `srtag:ReferentUser` (e.g. an archivist) who is in charge of building a consensual point of view, `srtag:TagStructureComputer` which corresponds to the software agents performing automatic handling of tags, and `srtag:ConflictSolver` corresponding to software agents which propose temporary conflict resolutions for diverging points of view before referent users choose one consensual point of view.

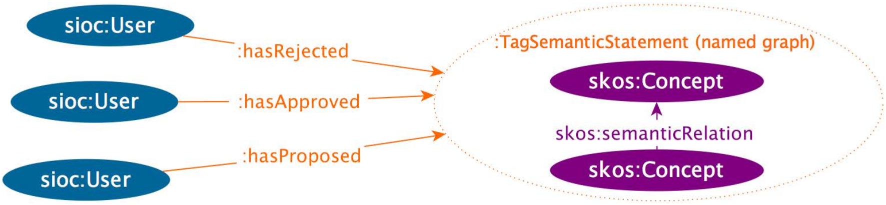


Fig. 1 SRTag RDF schema

3.2 Folksonomy Enrichment Life Cycle

As a result, our model allows for the factorization of individual contributions as well as the maintenance of a coherent view for each user and a consensual view linked to a referent user. Furthermore, by modeling different types of agents who propose, approve or reject tag relations, we are able to set up a complete life cycle of enriched folksonomies. Figure 2 illustrates this life cycle which starts with a "flat" folksonomy (ie. with no semantic relationships between tag) and can be decomposed as follows:

1. Automatic processing is performed on tags using methods based on an analysis of the labels of tags and on the network structure of the folksonomy. `srtag:-TagStructureComputer` agents then add assertions to the triple store stating semantic relations between tags . These computations are done overnight due to their algorithmic complexity.
2. Human agents, modeled as `srtag:SingleUser`, contribute through user friendly interfaces integrated into tools they use daily by suggesting, correcting or validating tag relations. Each user maintains his point of view, while benefitting from the points of view of other users.
3. As logical inconsistencies may arise between all users' points of view, another type of automatic agent (`srtag:ConflictSolver`) detects these conflicts and proposes resolutions. The statements proposed are used to reduce the noise that may hinder the use of our system when, for instance, different relations are stated about the same pair of tags.
4. The statements from the conflict solver agent are also used to help the referent user in her task of maintaining a global and consensual view with no conflicts. This view can then be used to filter the suggestions of related tags by giving priority to referent-validated tags over other tags suggested by computers.
5. At this point of the life cycle we have a semantically structured folksonomy in which each user's point of view co-exists with the consensual point of view. Then a set of rules is applied to exploit these points of view in order to offer a coherent navigation to all users.
6. Another cycle restarts with automatic handlings to take into account new tags added to the folksonomy.

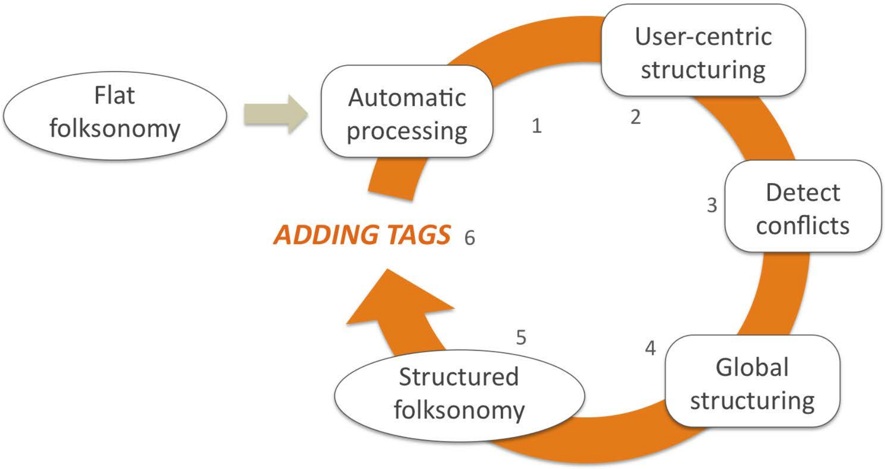


Fig. 2 Folksonomy enrichment lifecycle

4 Automatically Extracting Emergent Semantics

Several types of methods can be applied to folksonomies in order to retrieve semantic relationships between tags. We first present the experiment we conducted with real data from the Ademe agency to evaluate the performance of string-based methods and our proposal to combine them efficiently. Then we present our integration of state of the art algorithms [Markines et al., 2009, Mika, 2005] analyzing the structure of folksonomies.

4.1 Evaluating the Performance of String-Based Metrics

4.1.1 Overview of Existing String-Based Metrics

String based distance measures consider the character strings of the labels of tags to be compared. For instance, the Levenshtein [Levenshtein, 1966] distance metric was used in [Specia and Motta, 2007] to group spelling variant tags such as "new_york" and "newyork". To go further in the use of these cost effective methods, we conducted a benchmark to evaluate the ability of such metrics to retrieve other types of semantic relations such as *related* relation, or *narrower* or *broader* relation, also called *hyponym* relation. Hyponym relations reflect the relative degree of generality between two notions such as, *e.g,* in: "pollution" is broader than "soil pollution". Two notions are merely related in the other cases, as for instance "energy" and "electricity".

We have compared the similarity metrics implemented in the package SimMetrics[11] which give, for a pair of strings (s_1, s_2), a normalized value between 0 and 1, with a value of 1 meaning that both compared strings are most similar. The similarity metrics we compared fall into several categories: (a) edit distance based methods, which consider the set of operations needed to turn string s_1 into string s_2; (b) token-based methods, such as overlap coefficient, which decompose strings into tokens separated by white space; (c) methods using vector representations of strings such as the cosine similarity; and finally (d) other types of metrics such as QGram or Soundex metrics.

4.1.2 Benchmarking

We have manually constructed a test sample from the tags used at Ademe to index their documents and resources. This sample, which mixes freely chosen tags and tags chosen by the archivists, was divided into 4 sets of 22 pairs of tags (t_1, t_2), each set containing tag pairs which correspond to a semantic relation, namely: spelling variant, hyponym, related, and unrelated. These relations have been validated by one member of the Ademe's archivists team so that it reflects the knowledge of our user's domain.

[11] http://www.dcs.shef.ac.uk/~sam/stringmetrics.html

The Monge-Elkan metric is a hybrid metric based on edit distances which also decomposes strings into tokens, and uses a second metric to compare each token with all the others. For our experiment we used a series of 15 metrics and the combination of theses 15 metrics with the Monge-Elkan method, which makes a total of 30 different metrics.

Our benchmarking approach consists in using a sample of pairs of tags (mostly in French), manually constructed and validated by a human expert (from Ademe in our case), and which will serve as a reference. This sample was divided into 4 subsets, each subset containing 22 pairs of tags linked with one type of semantic relation, namely: spelling variant, hyponym, related, and unrelated. To evaluate the performance of each metric in retrieving the right relations from our sample, we have computed for each subset the recall, precision and the weighted harmonic mean F_1 (to give as much importance to recall as to precision). These values were computed varying the threshold above which a given tag pair is retrieved or not. Then to count the false positive and true positive pairs that were matched, we applied the following rules: (a) for the related case the true positives are counted from all subsets except the unrelated subset, since spelling variant and broader/narrower pairs can be considered also “related”; (b) for the spelling variant and hyponym case, the true positives were only those from their corresponding subset, and the pairs from all the other subsets were counted as false positives.

We have then looked at the best metric for each type of relation by ranking them according to the mean value and the statistical deviation of F_1. The outcome of this first evaluation is that the Monge-Elkan_Soundex method outperformed other metrics in the related case. The best in the spelling variant case is the Jaro-Winkler metric, and the best for the hyponym is the MongeElkan_NeedlemanWunch metric. In the latter case however, none of the top metrics clearly outperformed the others. We should also notice the greater deviation in the related case than in the two other cases, and this result was expected since the fact for two notions being related rarely translates to some terminological similarities e.g. "car" and "wheel" are related but don’t share any letters. Now we are interested in finding a way, using these metrics, to differentiate between the 3 types of semantic relations.

4.1.3 Identifying Different Types of Relations

Now we are interested in finding a way, using these metrics, to differentiate between the 3 types of semantic relations.

First, we use the MongeElkan_Soundex metric to retrieve all related tag pairs, that is, all pairs sharing a relation which is at least of type “related”, meaning that in this category we’ll retrieve also spelling variant and hyponym cases. To do that, we must determine a threshold of the similarity value from the MongeElkan_Soundex metric above which a pair is considered related. To determine this threshold, we looked at the mean similarity value for all related cases (spelling variant, hyponym, related) and for all unrelated cases in the sample set. The results are shown in

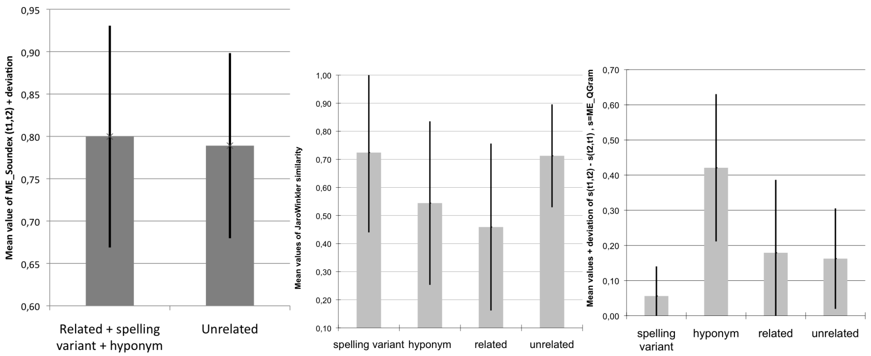


Fig. 3 (left) Comparison of the mean value of the MongeElkan_Soundex metric for all related cases (spelling variants, hyponyms and mere related) and for unrelated cases. (middle) Comparison of the mean value of the JaroWinkler metric for each type of semantic relation. (right) Mean value of the difference $\delta = s(t_1,t_2) - s(t_2,t_1)$ with s being the Monge-Elkan_QGram metric for each set of tag pairs.

fig. 3(left). We can see that, considering the deviations, if we choose a threshold value of 0.9 we are able to avoid unrelated pairs.

To distinguish spelling variant from related pairs, we look at the mean value and deviation of the best metric in the spelling variant case. In figure 3 (middle) we show the mean value of the JaroWinkler metric for the four types of semantic relations. We see that, taking into account the deviation, if we choose a threshold above 0.9 we are more likely to retrieve spelling variant pairs. This result is confirmed when we look at the threshold value for which F_1 is maximum for the JaroWinkler measure in the spelling variant case.

Next, we want to find a way to tell hyponym pairs from related pairs. The MongeElkan metrics are not symmetric, and we have calculated, for each tag pair (t_1,t_2), the difference $\delta = s(t_1,t_2) - s(t_2,t_1)$, with s being one of the 15 combination of MongeElkan with another metric. In figure 3(right) we give the mean value and deviation of δ for each set of tag pairs according to the MongeElkan_QGram metric, which performed best in this respect. We only included in this computation tag pairs that were retrieved thanks to the MongeElkan_Soundex metric and counted "related". We can see that if we choose a threshold above 0.39 (the highest value for δ when including the deviation), we are able to retrieve tags sharing a hyponym relation. When taking into account the sign of the difference, we are able to tell the direction of this relation, meaning that if we have δ negative and above a certain threshold, then t_1 can be considered narrower than t_2.

4.1.4 Heuristic String Based Methods (Algorithm 1)

As a result we are able to propose a heuristic (see algorithm14) that combines the best metrics to retrieve different semantic relations between tags. We first look for pairs of related tags (t_1,t_2) using Monge-Elkan_Soundex with a first threshold τ_a so that we have $s(t_1,t_2) \geq \tau_a$. This first threshold is chosen as explained in 4.1.3, ie $\tau_a = 0.9$ in our case. Then, we compare the JaroWinkler similarity with a second threshold τ_b to see whether the tags are spelling variants, such that $s(t_1,t_2) \geq \tau_b$. The threshold in this case is chosen as explained in 4.1.3, *i.e.* in our case, 0.94. If it's not the case, we use a third threshold τ_c and we compute the difference δ of the MongeElkan_QGram metric $\delta = s(t_1,t_2) - s(t_2,t_1)$, and if δ is such that $\delta \leq -\tau_c$, then we can infer that t_1 is narrower than t_2, or if $\delta \geq \tau_c$ then t_1 is considered broader than t_2. The third threshold is chosen after the results shown in figure 3 by picking a value above 0.39. In this process we give priority to the detection of spelling variants since string based methods are better suited for this type of relation, and by checking this case first we make sure to retrieve as many spelling variant cases as possible since those retrieved have statistically more chance to be true positive.

Algorithm 14. Heuristic string based metric to retrieve semantic relations between tags

```
for all distinct pair of tags $(t_i,t_j)$ from $S = \{t_1,t_2,...,t_n\}$ do
    if $MongeElkanSoundex(t_i,t_j) > \tau_a$ then
        if $JaroWinkler(t_i,t_j) > \tau_b$ then
            $t_i$ has spelling variant $t_j$
        else if $MongeElkanQGram(t_i,t_j) - MongeElkanQGram(t_j,t_i) \leq -\tau_c$ then
            $t_i$ has broader $t_j$
        else if $MongeElkanQGram(t_i,t_j) - MongeElkanQGram(t_j,t_i) \geq \tau_c$ then
            $t_j$ has broader $t_i$
        else
            $t_i$ has related $t_j$
        end if
    end if
end for
```

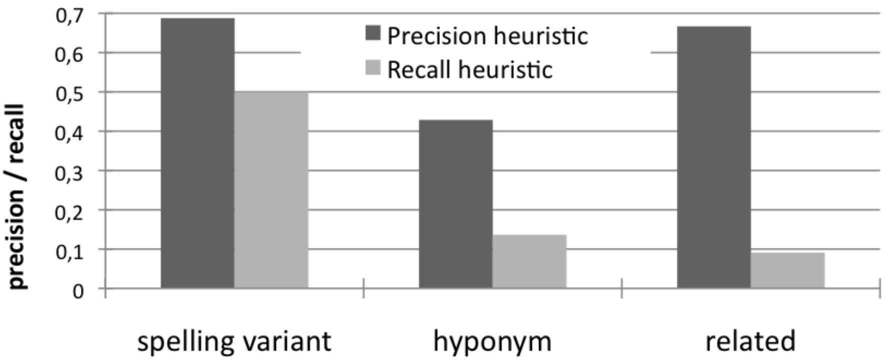


Fig. 4 Performance of the heuristic string-based metric (Algorithm 1)

We have applied our heuristic method to the same sample test. However, this heuristic is not directly comparable to the other metrics as it combines different methods and retrieves 3 types of semantic relations at a time, while in the global comparison experiment each metric was dealing with one type of semantic relation at a time. However, in order to evaluate quantitatively the global performance of this heuristic string-based metric, we show in figure 4 the values of the precision and recall for the 3 types of relations. We can clearly see in this figure that string based metrics perform best in the spelling variant case, which confirms a natural intuition since string-based methods were originally designed to match similar strings. Nonetheless, the noticeable performance in the hyponym case is explained with the ability of string-based metrics to easily detect common tokens such as in "pollution" and "soil pollution" and these cases often correspond to a hyponym relation. The related case is more difficult (hence the low precision) as this relation is the fuzziest and probably the least noticeable in the actual spelling of the tags ("sun" and "energy" *e.g*). Finally, this indicates the need to use other methods to be able to cover other cases where semantically related tags are not morphologically similar.

4.2 Analyzing the Structure of Folksonomies

In this section we detail our implementation of two methods (named Algorithm 2 and Algorithm 3 in the remaining) to extract emergent semantics that analyze the tri-partite structure of the folksonomy.

Algorithm 2

In order to extract *related* relationships between tags, we use the similarity measure based on distributional aggregation in the tag-tag context [Cattuto et al., 2008]. Cattuto *et al.* compared the different context in which similarity measures can be computed and studied the different type of semantic relationships they bring using the hierarchical structure of Wordnet. This experiment shows that tags associated via similarity measures based on simple co-occurrence tend to share subsumption relationships, whereas tags associated via distributional similarity measures in the tag-tag context tend to be on the same level of a semantic hierarchy, either having the same parents and grand-parents. Cattuto et al. explain that associating tags via their co-occurrence on a single resource accounts for their simultaneous use in the same act of tagging, where the user may have a tendency to span different levels of generality. For instance the tags "java" and "programming" are likely to be used simultaneously, and we can assume that they have, in the user's mind, different levels of generality. The relationship measured by the distributional measure based on the tag-tag context associates tags which share similar patterns of co-occurrence, but which are not necessarily or rarely used together. This is the case for example of the tags "java" and "python" which may be rarely used together, but each may be often used with the tag "programming".

To compute the tag-tag context similarity, we first consider the vector representation v_i of each tag t_i in this context. Each entry of this vector v_i is given by $v_{t_i t_j} = w(t_i,t_j)$ for $t_i \neq t_j$ where $w(t_i,t_j)$ corresponds to the co-occurrence value for the tags (t_i,t_j), and $v_{t_i t_i} = 0$. We set to zero the value for a tag with itself so that we consider tags to be related when they are found in a similar context, but not when co-ocurring together. The similarity value for a pair of tag (t_i,t_j) in the tag-tag context is then given by the cosine distance between the vectors v_i and v_j: $\cos(v_i,v_j) = \frac{v_i.v_j}{\|v_i\|_2 \cdot \|v_j\|_2}$. When this value is above a given threshold, we create an annotation saying that tag t_i is related to tag t_j.

Algorithm 3

In order to extract subsumption relations, we made use of the method described by [Mika, 2005] which consists in looking at the inclusions of the sets of users associated to a tag. Let S_i be the set of users using tag t_i, and S_j be the set of users using tag t_j. If the set S_i is included in the set S_J, so that we have $S_i \subset S_j$, with $card(S_i) > 1$ and $card(S_j) > card(Si)$, we can infer that the tag t_j is broader than the tag t_i.

Note that these two algorithms are not incremental since we have to analyze the whole folksonomy to compute the similarity of newly added tags.

4.3 Automatic Processing on a Real-World Dataset

We have performed the three types of calculation described above on a real-world dataset made of the following parts: (a) *delicious* dataset[12] comes from delicious.com and is made of the tagging of users who tagged at least one of their bookmarks with the tag “ademe” as of the 1st of October, 2009. (b) *thesenet* dataset comes from a database of Ademe which lists all the PhD projects funded by the agency. Each keyword has been considered as a tag, each identified project as a tagged resource, and each PhD student as the tagger. *(c) caddic* dataset is made of all entries of the past five year of the documents indexing base of the Ademe’s archivists. Each document corresponds to a tagged resource, and each keyword from the list of keywords associated to each document corresponds to a tag, with the archive service as the only tagger since no trace of the person who validated each entry is kept. In table 1 we detail, for each dataset: the number of distinct tags; the number of restricted tagging, *i.e.* the number of tripartite links between one resource, one tag and one user; the number of posts, *i.e.* the number of set of tags assigned by one user to a single resource (as a bookmark in delicious.com); the number of distinct tagged resources; and the number of users.

In table 2 we give some details on the results we obtained for each of the three methods of computation (Algorithm 1, 2, and 3) when applied to the three datasets. The first thing to notice is that algorithm 1 yields far more results (71034 statements)

[12] This subset of our experimental data is availlable, as of the time of writing, on the Isicil website `http://isicil.inria.fr`

Table 1 Description of the dataset

	delicious	thesenet	caddic	Full Dataset
Nb. distinct Tags	1015	6583	1439	9037
Nb. Restricted Tagging (1R - 1T - 1U)	3015	10160	25515	38690
Nb. distinct Resources	196	1425	4765	6386
Nb. distintc Users	812	1425	1	2238

Table 2 Description of the results of automatic processing

	Algo. 1	Algo. 2			Algo. 3		Total
	Full dataset	delicious	thesenet	caddic	delicious	thesenet	
Nb. related	59889	8141	206	30	-	-	68633
Nb. Broader/Narrower	10952	-	-	-	106	196	11254
Nb. Spelling variants	3193	-	-	-	-	-	3193
Computation time (s)	20952	4200	180	300	5	10	25647
Total number of statements							83080
Nb. of pairs with overlapping statements between different methods							31
Nb. of pairs with conflicting statements between different methods							22
Total number of statements on distinct pairs							83027

than algo. 2 (8377 statements in total, with 97% from delicious dataset). The results for algorithm2 can be explained because this method looks at the pattern of co-occurrence of tags, and delicious is the dataset in which two tags are more likely to have similar patterns of co-occurrence since, if we look at the ratio between the number of restricted tagging over the number of distinct resources, we obtain 15.38 for delicious, 7.13 for thesenet, and 5.35 for caddic. This says that there are more than twice as much distinct users who tagged the same resource in delicious as in thesenet or caddic. In addition, in delicious, a greater number of users tag the same resource using a smaller set of distinct tags, hence the greater probability for two tags to have similar patterns of co occurrence. For algorithm 3 we obtained a greater number of relations in the thesenet dataset than in the delicious dataset since the thesenet dataset has around 75% more users and even more distinct tags (around 6 times as many), hence a greater probability of having embedded sets of users of common tags.

In the bottom part of table 2 we see that, in total, we obtained 83080 statements from the 3 types of computation applied on our 3 datasets. Few of these statements (31) overlap with each other, i.e. some of them state identical relations between a given pair of tags as other statements established by another method of computation. Likewise, a few statements (22) contradict statements from different methods on the same pair of tags. After removing overlapping and contradictory statements, we obtain a total of 83027 statements.

This automatic handling is performed during low activity periods due to their algorithmic complexity, and each resulting statement is linked to the corresponding type of agent, each modeled as a subclass of `srtag:AutomaticAgent`.

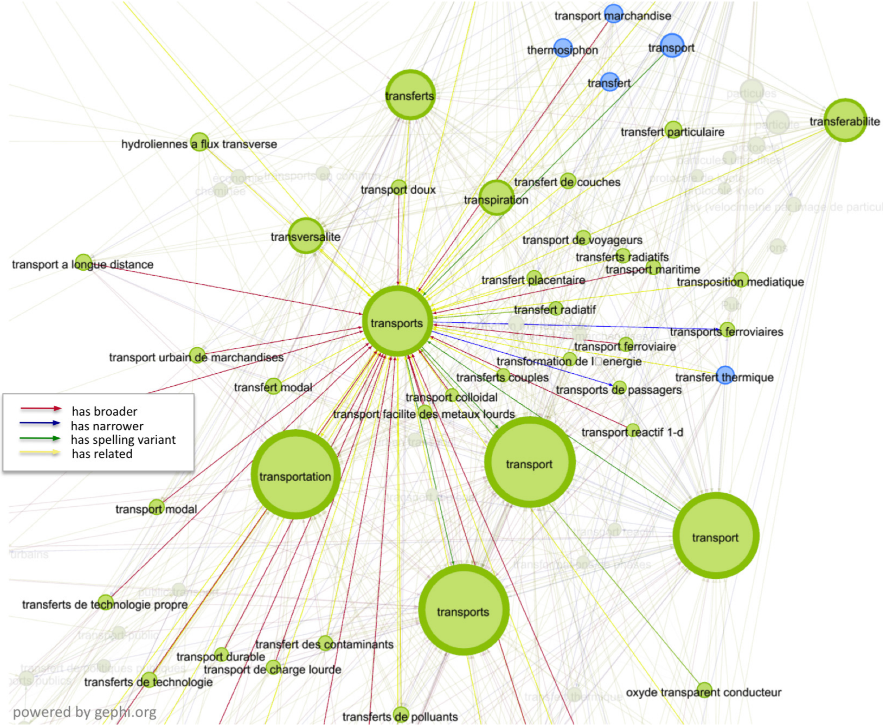


Fig. 5 Example of the results of automatic processing with the String Based method showing tags linked with the tag “transports”. The size of the nodes indicates the number of entering edges (in degree). The green nodes correspond to tags from thesenet and delicious dataset (hence the two nodes “transport”), and blue nodes correspond to tags from caddic dataset.

Moreover, algorithm 2 and algorithm 3 are not incremental since when new tags are added, the structure of the whole folksonomy is modified. This is not the case for algorithm 1 that only compares the labels of newly added tags with all the other tag labels. To give an example of the computation time, the total time to apply this 3 methods on the full dataset is 25647s in our setup, with a machine equipped of a 4 core Intel Core2 Duo processor running at 3.00 GHz with 8Go of RAM. In figure 5 we give an example of the results obtained with the String Based method for the tag “transports”.

5 Capturing and Exploiting Individual Contributions

Up to this point we have presented the different methods of computing tag relations and the model, SRTag, to keep track of the diverging points of view from all users. Now we are going to see how these points of view are first captured, then sorted out and arranged together in a coherent system.

5.1 Capturing Users Contributions

Once we are able to support diverging points of view, we want to allow users to contribute to the semantic structuring of the folksonomy while keeping as low as possible the cognitive overhead that this task may involve. To achieve this goal we propose integrating simple and non-obtrusive structuring functionalities within everyday user tasks. For instance, in our target community at Ademe, we want to be able to capture the expertise of the engineers when they browse the corpus of Ademe resources.

The design of the solution we propose is grounded on previous studies and development of collaborative ontology editors conducted in our research team (see [Peron, 2009] for a synthesis). Indeed, these studies set a background of considerations and evaluations regarding the ergonomic aspect of tools allowing the collaborative editing of a shared knowledge representation such as an ontology (ECCO[13]) or a structured folksonomy (SweetWiki by [Buffa et al., 2008]). The ergonometric analysis of the folksonomy editor of SweetWiki revealed several weaknesses that we tried to overcome in our proposal for an interface to capture users contributions regarding the semantics of tags. By taking into account the multiple points of view we make sure that (1) each user is not reluctant to contribute because of a fear to destroy others' contributions, and (2) each point of view is kept in order to obtain a richer knowledge representation in the end.

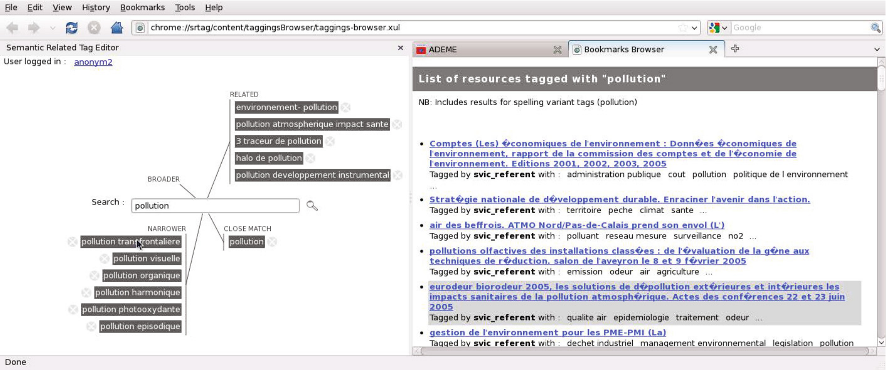
Our proposal consists in an interface for explaining the computed structure of the folksonomy in which tags are suggested and ordered according to their semantic relations with the current searched-for tag (see figure 6). Related and spelling variant tags are positioned on the right side (respectively top and bottom corner) and broader and narrower tags are positioned on the left side (respectively top and bottom corner). Optionally, users can either merely reject a relation by clicking on the cross besides each tag, or they can correct a relation by dragging and dropping a tag from one category to another.

5.2 Detecting and Solving Conflicts

5.2.1 ConflictSolver Mechanism

A third type of agent is introduced, modeled with a subclass of `srtag:-AutomaticAgent` named `srtag:ConflictSolver` and which looks for conflicts emerging between all user's points of view. A conflict in the structured folksonomy emerges when different relations have been proposed or approved by different users on the same pair of tags (if a user changes his mind, we simply update his point of view). For instance, the tag "pollution" is narrower than "co2" for a number n_1 of users, but for a number n_2 of users "pollution" is broader than "co2".

[13] French for Collaborative and Contextual Ontology Editor, see `http://www-sop.inria.fr/edelweiss/projects/ewok/publications/ecco.html`



List of resources tagged with "pollution"

NB: Includes results for spelling variant tags (pollution)

- Comptes (Les) �conomiques de l'environnement : Donn�es �conomiques de l'environnement, rapport de la commission des comptes et de l'�conomie de l'environnement. Editions 2001, 2002, 2003, 2005
  Tagged by **svic_referent** with : administration publique cout pollution politique de l environnement ...
- Strat�gie nationale de d�veloppement durable. Enraciner l'avenir dans l'action.
  Tagged by **svic_referent** with : territoire peche climat sante ...
- air des beffrois. ATMO Nord/Pas-de-Calais prend son envol (L')
  Tagged by **svic_referent** with : polluant reseau mesure surveillance no2 ...
- pollutions olfactives des installations class�es : de l'�valuation de la g�ne aux techniques de r�duction. salon de l'aveyron le 8 et 9 f�vrier 2005
  Tagged by **svic_referent** with : emission odeur air agriculture ...
- eurodeur biorodeur 2005, les solutions de d�pollution ext�rieures et int�rieures les impacts sanitaires de la pollution atmosph�rique. Actes des conf�rences 22 et 23 juin 2005
  Tagged by **svic_referent** with : qualite air epidemiologie traitement odeur ...
- gestion de l'environnement pour les PME-PMI (La)
  Tagged by **svic_referent** with : dechet industriel management environnemental legislation pollution

Fig. 6 Firefox extension seamlessly integrating tag structuring capabilities (left part). The user was about to drag the tag "energy" towards the "spelling variant" area to state that the tag "energie" (the tag currently searched for) is a spelling variant of "energy". On the right side are displayed the resources associated to the current tag.

In addition, other users can say that "pollution" is related to "co2". In this case the conflict solver first counts the number of approval $nbApp_i$ for each conflicting statement $s_i \varepsilon \{s_i\}_n$, n being the total number of statements made on a given pair of tags. Then, it retrieves the maximum $max\{nbApp_i\}_{i\varepsilon[1,n]} = nbApp_{max}$, and compares the ratio $r = \frac{nbApp_{max}}{\sum_n nbApp_i}$ with a given threshold τ_{cs}. If this ratio is above τ_{cs}, then the conflict solver approves the corresponding statement. Otherwise, if r is below τ_{cs}, this means that no strong consensus has been reached yet, and the conflict solver merely says that both tags are related since this relation is the loosest and represents a soft compromise between each diverging point of view. In this case it approves the related statement if it exists, and if not, it proposes its own related statement.

5.2.2 Experiment

Protocol

We have conducted an experiment among 5 members of Ademe. We have presented them with a list of 94 pairs of tags (t_1, t_2) and asked them to choose a semantic relation between t_1 and t_2 among the following: t_1 is a spelling variant of t_2, t_1 is broader than t_2, t_1 is narrower than t_2, t_1 is related to t_2, or t_1 is not related to t_2. We have then applied the conflict solver on the set of relations and points of view. When a user chose the fifth possibility, *i.e.* that t_1 is not related to t_2, we have applied a SPARQL rule to translate this choice into the rejection of all the relations (namely spelling variant, broader, narrower, and related) stated about the same pair of tags. Doing this allows us to consider relations that are debatable, in the sense that some users have approved it, and some other users have rejected it, but none have proposed or approved another relation.

After applying the conflict solver, we are able to distinguish between 4 cases regarding the relation between two tags:

1. Approved statements: when a relation has only been approved.
2. Conflicting statements: when some users have proposed a relation and some other users have approved another relation on the same pair of tags, e.g. some users have approved that "pollution" has broader "pollutant", and some other users have approved that "pollution" has spelling variant "pollutant".
3. Debatable statements: when only one relation is stated on a given pair of tags but this relation has been both approved by some users and rejected by some others.
4. Rejected statements: when a relation has only been rejected.

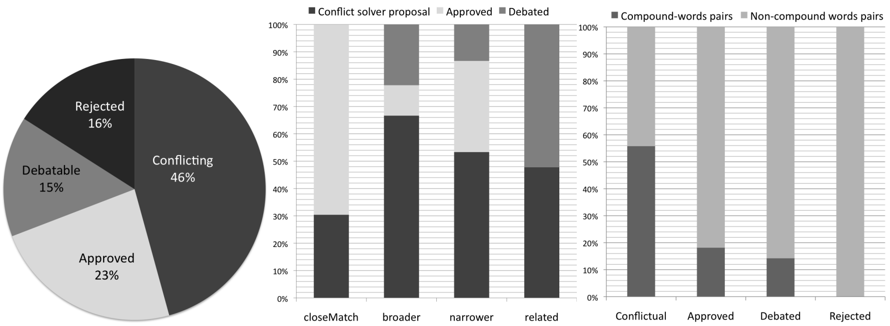


Fig. 7 Result of conflict solving. (a) Distribution of the different cases of conflict solving for all pairs of tags. (b) Distribution of the different cases of conflict solving for each type of semantic relations. (c) Distribution of pairs with compound words compared with pairs with non-compound words for each type of conflict solving cases.

Result analysis

In figure7 we show the detailed results of the conflict solver applied on our dataset gathered from the 5 users who chose one relation for each of the 94 pairs of tags of the dataset. The first chart (a) shows the distribution of the different cases of conflict solving over the 94 pairs of tags. We see that for almost half of the pairs (46%), users proposed several relations for a single pair (Conflicting case).

Then in the second chart (b) we looked at the distribution of conflict solving cases for each type of semantic relation. Since several relations are stated in the conflicting case, we kept only in this chart the relations that were proposed by the conflict solver, i.e. the relations that were supported by a clear majority or proposed as a compromise. We see in this chart that 70% of the close match statements were only approved by users, and that 30% were proposed by the conflict solver. If we look at the broader and narrower case altogether (since these relations are the inverse of each other), we see that they are involved in conflicts in more than 50% of the cases. Lastly, the related relation has never been only approved by users and is either involved in conflicts (48% of the statements) or is debatable (52% of the statements).

We should note here that “related” has been proposed as a compromise without being approved by any user once and gained a clear majority 3 times out of the 43 cases of pairs with conflicts. This means that in most of the cases where “related” is proposed by the conflict solver, this relation serves as a compromise between proposals of other relations. Thus, chart (b) shows that the “close match” is the relation that is the most capable of bringing an explicit consensus, and it is clear that it is easier to agree on the fact that “ecology” and “ecologie” refer to the same notion, than it is to agree on saying that “collective action” is narrower than “collectivity”. Indeed, both tags in the latter case may not directly be related to all users mind, and moreover, the type of relation that these two tags share is disputable and strongly depends on the level of expertise of the user who is to choose a relation (some users with a high level of expertise in the corresponding field will be willing to neatly articulate both notions, maybe opting for broader or narrower, while some other less expert users will simply be willing to account for the fact that there is a relation with “related”, or will even be ready to merge both notions because they are not too concerned about the distinctions that can be made).

In the third chart (c) we examined the influence of another noticeable feature that may distinguish different types of pair of tags. Some pairs of tags consist of a word for the first tag and a compound word for the second tag made of the first tag (as in “pollution” and “soil pollution”) or one of its derivative (as in “pollution” and “pollutants detection”), and this concerns 30 pairs out of 94. In this chart we plotted the distribution between two types of pairs of tags, i.e. pairs with compound words and the rest of the pairs, for each case of conflict solving. The result shows that conflicting pairs are pairs with compound words in the majority of the cases (56%). Likewise, only 18% of the only approved statements and 14% of debatable statements (we recall that in this case only one relation is stated, though it can be approved and rejected) were involving pairs with compound words, and this type of pairs was never at the origin of only rejected statements. This suggests that pairs with compound words are more likely to cause conflicts, and rarely lead to clear consensuses.

5.3 Creating a Consensual Point of View

The fourth type of agent we introduced is the `ReferentUser`. The referent user will be able to approve, reject or correct all the relations already existing in the structured folksonomy in order to maintain its own and consensual point of view. The conflict solver mechanism will assist the referent user in her task by pointing out the conflicts already existing in the collaboratively structured folksonomy. Then, all the statements that the referent user has already treated will be ignored in further passes of the `ConflictSolver`. The consensual point of view can be used to generate a hierarchical tag cloud from the folksonomy where broader tags are printed in bigger fonts than narrower tags. This type of tag cloud may be useful to guide the users in giving him a panoramic view of the content of the folksonomy and can be presented at a starting point of the navigation, indicating the broadest tags, and then, during the search, giving the semantic surrounding of the current tag by showing broader and narrower tags.

5.4 Exploiting and Filtering Points of View

At this stage of the process, we obtain a folksonomy semantically structured via several points of view, among which a global and consensual point of view emerges. We present in this section the strategies we propose for exploiting these points of view in order to present a coherent experience to all users of the system.

By keeping track of the type of agents associated to each statement, we are able to give a priority to the suggested tags corresponding to these statements when a user u searches for a tag t. The system issues 5 SPARQL queries looking for statements made on the searched-for tag and each time approved by different types of user but making sure these statements do not conflict with preceding results. All results will then be merged and used to suggest tags semantically related to t. The priority order followed is given below:

1. all statements S_u approved by the user u.
2. all statements S_{ru} approved by the `ReferentUser`, except if they conflict with one from S_u.
3. statements S_{cs} approved by the `ConflictSolver`, except if they conflict with one from S_u or S_{ru}.
4. all statements S_{ou} approved by other users, except if they conflict with one from S_u, S_{ru}, or S_{cs}
5. all statements S_{tc} approved by the `TagStructureComputer`, except if they conflict with one from S_u, S_{ru}, S_{cs}, or S_{ou}.

This set of rules allows, when suggesting tags to a user during a search, filtering out the conflicting or more general points of view from the other contributions, coming from humans or machines. For example, if the user is searching for the tag "energy", the system will first suggest tags coming from assertions she has approved, e.g. if current user has approved that "energies" is a spelling variant of "energy", it will suggest "energies". We give an example in listing 7.1 of the second query that is issued on named graphs and that looks for statements approved by the `ReferentUser` (line 1 to 4) and that (*i*) are not rejected by current user (line 5 to 8) and (*ii*) that do not conflict with the ones approved by the current user (line 9 to 13). For instance if the `ReferentUser` had approved that "energies" has broader "energy", this assertion will not be included in the results since, in the SRTag ontology, the property `skos:closeMatch` (this is the property we use for spelling variants) is declared to be `srtag:incompatibleWith` the property `skos:broader`. The system proceeds with the next queries, following the priority order described above. As a consequence, it allows each user to benefit from the other users contributions while preserving a coherent experience using a referent point of view or, when absent, using the point of view of the conflict solver.

Listing 7.1 SPARQL query used to retrieve statements about the tag “energy” and approved by the `ReferentUser` but not directly rejected by the current user or contradictory with statements he has approved.

```
SELECT * {
GRAPH ?g {?search-tag ?p ?suggested-tag}
FILTER(?search-tag = <http://ex.org/tag/energy>)
?g rdf:type srtag:ReferentValidatedStatement
OPTIONAL {
    ?u srtag:hasRejected ?g
    FILTER(?u  = <http://ex.org/users/me>)}
FILTER(!bound(?u))
OPTIONAL{
    GRAPH ?g2 {?search-tag ?p2 ?suggested-tag}
    ?g2 srtag:approvedBy <http://ex.org/users/me>
    ?p srtag:incompatibleWith ?p2   }
FILTER (!bound(?g2)) }
```

6 Conclusion and Discussion

In this paper, we presented our approach to the semantic enrichment of folksonomies. We propose a socio-technical system in which automatic agents help users in maintaining their personal points of view while still benefiting from others’ contributions, and also helping referent users in their task of building a consensual point of view. Our approach is grounded on a careful usage analysis of our target communities that allows us to include their daily activity in the process.

In order to bootstrap the process, we make use of the automatic handling of folksonomies to extract the emergent semantics. In this regard, we proposed in this paper an evaluation of the main string-based methods. in order to: (a) motivate the choice of the metrics performing best in our context; and (b) evaluate the ability of such metrics to differentiate the semantic relations typically used in thesaurus, *i.e.* to be able to tell when two tags are merely related, or when one tag is broader or narrower than another tag, or when two tags are spelling variants of the same notion. As a result we proposed a heuristic metric that performs this task. This heuristic metric performs best for detecting spelling variants, as expected. The values of the thresholds for this method are chosen after a calibration phase conducted with the help of several Ademe’s agents. Therefore, further studies are required in order to validate the robustness and the sensibility to the threshold values but the objective of this work was to check wether or not string-based distances are relevant to detect other relations that spelling variant, and we have shown here promising results for subsumption relations in cases such as “pollution” which is broader than “soil pollution”.

We have also quantitatively shown that the approaches analyzing the structure of folksonomies are necessary to retrieve semantic relations when tags sharing semantic relations are not morphologically similar, even if they are more costly and not incremental, and we have presented the results of these three types of method that we obtained on a real world dataset.

In order to capture diverging points of view in the semantic structuring of folksonomies, we proposed a formal ontology that makes use of named graphs to describe semantic relations between tags. The points of view of users are then attached to these asserted relations. By describing the different classes of agents who propose or reject asserted relations, we are able to model a complete life cycle for a collaborative and automatically assisted enrichment of folksonomies. (1) This cycle starts with a flat folksonomy which is first analyzed by automatic agents which propose semantic relations. (2) The users can contribute and maintain their own point of view by validating, rejecting, or proposing semantic relations thanks to a user-friendly interface integrated in a navigation tool. (3) The conflicts emerging from these points of view are detected and (4) utilized to help a referent user to maintain a global and consensual point of view. (5) The result of this process is a folksonomy augmented with semantic assertions each linked to different points of view coexisting with a consensual one. (6) The cycle restarts when new tags are added or when relations are suggested or changed. Semantic assertions are used to suggest tags when navigating the folksonomy, and a set of formal rules allows filtering the semantic assertions in order to present a coherent experience to the users while allowing them to benefit from others' contributions.

Our approach is currently being tested at the Ademe agency to enhance the browsing of its online corpus available to members of the agency and to the public. These tests will also help us to improve the user-friendliness of our interface to browse semantice relationships. In this context the expert engineers of Ademe maintain their points of view so as to reflect on their expertise in a given domain. At the same time, the archivists (our referent users) are assisted in the task of enriching with new tags and semantically structure their global point of view from the collaborative enrichment of the folksonomy.

Our future work includes testing our approach with the users of Ademe. We also plan on exploiting the semantic relations between tags at tagging time to guide and help users provide for more precise tags, and also to provide for additional input material for semantic social network analysis [Ereteo et al., 2009]. We plan in this respect to propose a novel approach to indexing where users and professional indexers, such as the Ademe's archivists, are engaged in a fruitful collaboration leveraged by a tailored automated assistance.

Acknowledgements. This work was funded by French National Agency for Research (ANR) under the reference ANR-08-CORD-011-05.

References

[Angeletou et al., 2008] Angeletou, S., Sabou, M., Motta, E.: Semantically Enriching Folksonomies with FLOR. In: CISWeb Workshop at European Semantic Web Conference ESWC (2008)

[Aussenac-Gilles et al., 2000] Aussenac-Gilles, N., Biébow, B., Szulman, S.: Corpus analysis for conceptual modelling. In: 12th International Conference Workshop on Ontologies and Texts at Knowledge Acquisition, Modeling and Management, EKAW 2000 (2000)

[Bouquet et al., 2004] Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., Stuckenschmidt, H.: Contextualizing ontologies. Web Semantics: Science, Services and Agents on the World Wide Web 1(4), 325–343 (2004); International Semantic Web Conference 2003

[Braun et al., 2007] Braun, S., Schmidt, A., Walter, A., Nagypál, G., Zacharias, V.: Ontology maturing: a collaborative web 2.0 approach to ontology engineering. In: CKC. CEUR Workshop Proceedings, vol. 273. CEUR-WS.org. (2007)

[Buffa et al., 2008] Buffa, M., Gandon, F., Ereteo, G., Sander, P., Faron, C.: SweetWiki: A semantic Wiki. J. Web Sem., Special Issue on Web 2.0 and the Semantic Web 6(1), 84–97 (2008)

[Carroll et al., 2005] Carroll, J.J., Bizer, C., Hayes, P., Stickler, P.: Named graphs, provenance and trust. In: WWW 2005: Proceedings of the 14th International Conference on World Wide Web, pp. 613–622. ACM, New York (2005)

[Cattuto et al., 2008] Cattuto, C., Benz, D., Hotho, A., Stumme, G.: Semantic Grounding of Tag Relatedness in Social Bookmarking Systems. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 615–631. Springer, Heidelberg (2008)

[Ereteo et al., 2009] Erétéo, G., Buffa, M., Gandon, F., Corby, O.: Analysis of a Real Online Social Network Using Semantic Web Frameworks. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 180–195. Springer, Heidelberg (2009)

[Gandon et al., 2007] Gandon, F., Bottolier, V., Corby, O., Durville, P.: Rdf/xml source declaration, w3c member submission (2007), http://www.w3.org/Submission/rdfsource/

[Golebiowska, 2002] Golebiowska, J.: Exploitation des ontologies pour la memoire d'un projet-vehicule - Methode et outil SAMOVAR. PhD thesis, Universite de Nice-Sophia Antipolis (2002)

[Heymann and Garcia-Molina, 2006] Heymann, P., Garcia-Molina, H.: Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems. Technical report, Stanford InfoLab (2006)

[Huynh-Kim Bang et al., 2008] Huynh-Kim Bang, B., Dané, E., Grandbastien, M.: Merging semantic and participative approaches for organising teachers' documents. In: Proceedings of ED-Media 2008 ED-MEDIA 2008 - World Conference on Educational Multimedia, Hypermedia & Telecommunications, Vienna France, pp. 4959–4966 (2008)

[Levenshtein, 1966] Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions and reversals. Soviet Physics Doklady 10(8), 707–710 (1966)

[Limpens et al., 2009] Limpens, F., Monnin, A., Laniado, D., Gandon, F.: Nicetag ontology: tags as named graphs. In: International Workshop in Social Networks Interoperability, Asian Semantic Web Conference 2009 (2009)

[Markines et al., 2009] Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., Stumme, G.: Evaluating similarity measures for emergent semantics of social tagging. In: 18th International World Wide Web Conference, pp. 641–641 (2009)

[Mika, 2005] Mika, P.: Ontologies Are Us: A Unified Model of Social Networks and Semantics. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 522–536. Springer, Heidelberg (2005)

[Passant and Laublet, 2008] Passant, A., Laublet, P.: Meaning of a tag: A collaborative approach to bridge the gap between tagging and linked data. In: Proceedings of the WWW 2008 Workshop Linked Data on the Web (LDOW 2008), Beijing, China (2008)

[Peron, 2009] Peron, S.: Etude ergonomique de folkon. Technical report, UNSA, INRIA (2009)

[Ribière, 1999] Ribière, M.: Représentation et gestion de multiples points de vue dans le formalisme des graphes conceptuels. PhD thesis, Université Nice-Sophia Antipolis (1999)

[Specia and Motta, 2007] Specia, L., Motta, E.: Integrating Folksonomies with the Semantic Web. In: Franconi, E., Kifer, M., May, W. (eds.) ESWC 2007. LNCS, vol. 4519, pp. 624–639. Springer, Heidelberg (2007)

[Tanasescu and Streibel, 2007] Tanasescu, V., Streibel, O.: Extreme tagging: Emergent semantics through the tagging of tags. In: Haase, P., Hotho, A., Chen, L., Ong, E., Mauroux, P.C. (eds.) Proceedings of the International Workshop on Emergent Semantics and Ontology Evolution (ESOE 2007) at ISWC/ASWC 2007, Busan, South Korea (2007)

[Van Damme et al., 2007] Van Damme, C., Hepp, M., Siorpaes, K.: Folksontology: An integrated approach for turning folksonomies into ontologies. In: Bridging the Gep between Semantic Web and Web 2.0 (SemNet 2007), pp. 57–70 (2007)

Ontology-Based Formal Specifications for User-Friendly Geospatial Data Discovery

Ammar Mechouche, Nathalie Abadie, Emeric Prouteau, and Sébastien Mustière

Abstract. Nowadays, a huge amount of geodata is available. In this context, efficient discovery and retrieval of geographic data is a key issue to assess their fitness for use and optimal reuse. Such processes are mainly based on metadata. Over the last decade, standardization efforts have been made by the International Standard Organization (ISO) and the Open Geospatial Consortium (OGC) in the field of syntactic interoperability between geographic information components. Among existing standards, ISO-19115 standard provides an abstract specification of metadata for geospatial data discovery and exploration, ISO-19110 defines feature catalogues, and ISO-19131 defines data product specifications. Yet, information provided by these standardized metadata is not always formalized enough to enable efficient discovery, understanding and comparison of available datasets. Notably, information regarding geodata capture rules can be represented only through free text into the standard metadata (ISO 19131). More generally, geospatial database capture specifications are a very complex but very rich source of knowledge about geospatial data. They could be used with benefits in geospatial data discovery process if they would be represented in a more formal way than in existing ISO standards. In this context, we propose a model to formalize geospatial databases capture specifications. Firstly, we extend existing standards to represent information that was insufficiently formalized. Second, we propose an OWL representation of our global model. This ontological approach enables to overcome the limitations related to string-based queries and to provide efficient access to data capture information. Finally, we implement this model as a Web application allowing users to discover the available geospatial data and to access to their specifications through a user-friendly interface.

Ammar Mechouche · Nathalie Abadie · Sébastien Mustière
Institut Géographique National, Laboratoire COGIT 73, Avenue de Paris,
94160 Saint-Mandé, France
e-mail: Ammar.Mechouche@gmail.com,
{Nathalie-f.Abadie,Sebastien.Mustiere}@ign.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 151–176.
DOI: 10.1007/978-3-642-35855-5_8

1 Introduction

Geographic vector databases aim at representing the real geographical space. They provide an abstract, partial, and not unique view of the geographical realm: geographic entities of the real world are represented in a simplified way, according to the database producer's point of view and to the database intended use. Recent progresses in the field of digital geospatial data acquisition combined with a growing development of Spatial Data Infrastructures (SDIs) and their associated standards aiming at facilitating interoperability between geographic information components have opened access to a plethora of different and complementary geodata sources, which are useful for many applications.

In order to avoid geodata misuse, users need to be aware of all assumptions and limitations related to the data conceptualisation and capture process, i.e. the database specifications. Therefore, such information about geodata specifications must be documented and provided to users to enable them to assess whether a given geodataset is really applicable or not for their intended use. This step of geodata search and applicability evaluation, also known as geodata discovery, is highly important to guaranty their efficient and consistent reuse.

Many standards have been developed by the International Standard Organization (ISO) and the Open Geospatial Consortium (OGC) in the field of syntactic interoperability between geographic information components. Notably, standards dedicated to geodata discovery and use have been published, and are currently used by geodata cataloguing systems. However, they do not consider sufficiently documenting geodata specifications. Consequently, users can rarely access to this very rich source of knowledge about geodata.

Recent works in the field of geodata discovery mainly focused on solving semantic heterogeneity problems due to keyword-based search through catalogues. The proposed approaches are generally inspired by federated databases or mediators [Sheth and Larson, 1990, Wiederhold, 1992] architectures developed in the field of computer sciences for solving semantic heterogeneity problems between heterogeneous information sources.

In this context, we propose a model to formalize geographic database specifications. Consistently with recent works in the field of geodata discovery, we extend existing standards to explicit geodata specifications thanks to expressive mappings linking local data source ontologies with a common global ontology. For the sake of genericity, these mappings –like their associated ontologies– are written in the recommended Ontology Web Language, namely OWL. This ontological approach enables to overcome the limitations related to string-based queries and to provide efficient access to data capture information. Finally, we implement this model in a Web application allowing users to discover the available geospatial data and to access to their specifications through a user-friendly interface.

The remainder of this article is structured as follows. First, we describe geodata discovery context, and the limitations of commonly used solutions. Then, we present some related works, both in the field of geodata discovery and in the field of formalisation of geographic database specifications. After that, we detail our ontological model and its implementation. Finally, we give some perspectives to our work.

2 Geodata Discovery Issues

2.1 Catalogues for Geodata Discovery

Geodata discovery is practically based on catalogues. Catalogues aim at indexing and describing available geodatasets in order to provide to users an efficient and user-friendly access to information about several aspects of each registered geodataset, such as theme, geographic extent, reference system, quality or genealogy, and also to geodatasets themselves. This information about geodatasets is made available as part of metadata, which are used as a consistent and structured information source. Metadata are queried by the search engine of the catalogue through keywords provided by users looking for specific geodatasets. Query' results are displayed through the interface of the catalogue to help users in discovering and understanding what kinds of data are available, and which of them really fit their needs.

Standardisation efforts made by the Technical Committee 211 dedicated to geographic information of the International Standard Organization (ISO), and the Open Geospatial Consortium (OGC) in the field of syntactic interoperability between geographic information components have lead to the publication of a comprehensive set of standards recommended to describe geodatasets. ISO-19115 standard [ISO, 2003] provides an abstract specification of metadata for geospatial data discovery and exploration. Its practical implementation is consolidated by ISO-19139 standard [ISO, 2007b] which defines XML encoding rules for ISO-19115 metadata. ISO-19115 standard focuses particularly on metadata related to the identification, the extent, the quality, the spatial and temporal schema, the spatial reference system and the distribution of a given geodataset.

Besides standards entirely dedicated to data discovery and exploration, ISO TC-211 defines standards about the use of geodata. Among them, ISO-19110 standard [ISO, 2005b] defines the methodology for cataloguing feature types available in a specific geodataset. It is based on ISO-19109 standard [ISO, 2005a] which focuses on application schemas description. Therefore, ISO-19110 standard describes geographic features at the feature type level and as such it provides information about geodatasets at a more precise level of granularity. Geodata implementing rules are described by the ISO-19131 standard [ISO, 2007a] on data product specifications. This standard provides information on the requirements that a given dataset should fulfill.

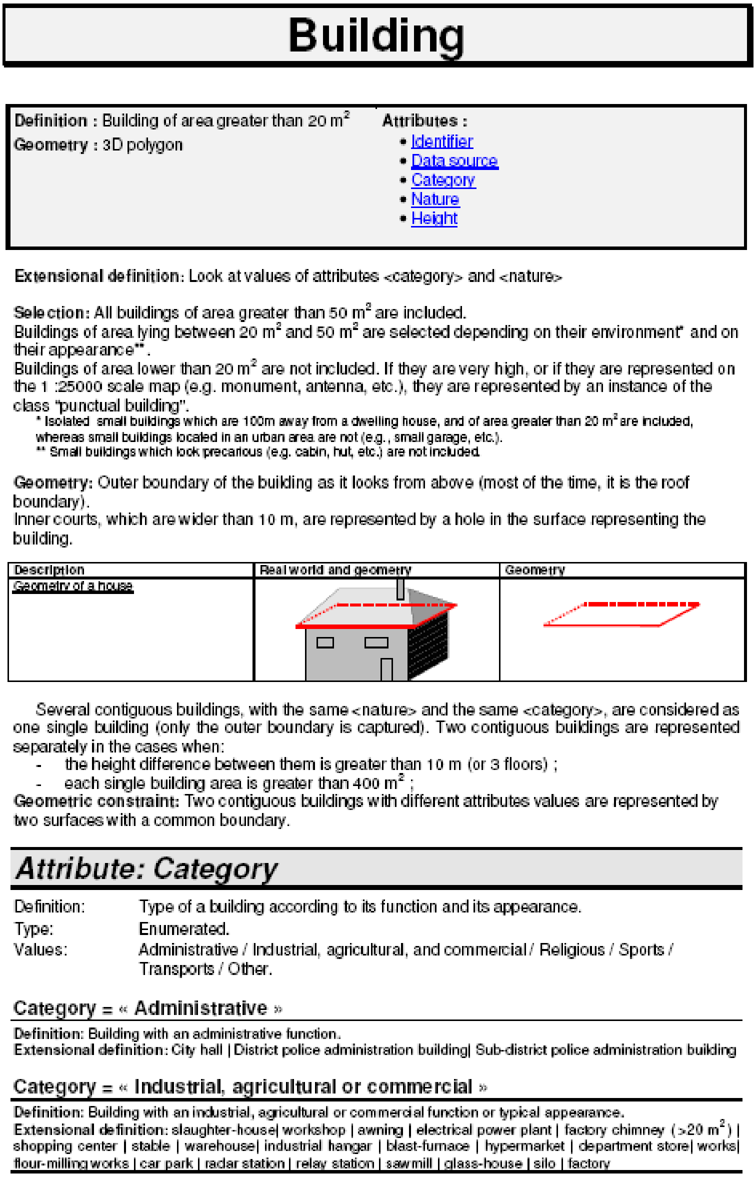
As required by INSPIRE implementing rules, most of geodata cataloguing tools implement ISO-19115 standard for metadata modeling. Therefore, information required by this standard is made available for users who want to discover geodatasets registered in any INSPIRE compliant catalogue. Information about the use of geodata can also be given to users provided that standards such as ISO-19110 and ISO-19131 are also used as metadata sources in the catalogue. However, even if they provide a lot of very useful information, these standards still lack of precision in the geodatasets description.

2.2 *Limitations of Catalogue-Based Geodata Discovery*

Despite these standardisation efforts, problems still arise notably when dealing with metadata semantics. Firstly, catalogue-based discovery of geodata is performed through keyword-based queries on metadata. However, problems of semantic heterogeneity due to natural language ambiguity and string-based queries cause limitations when searching information in metadata: users can miss useful information due to synonymy, homonymy or simply typographic mistakes problems.

In addition to that, existing standard metadata do not enable to formalise enough complex information about geodata capture process. Actually, each geodata producer has its own rules for data capture, and its own point of view about the geographical real world [Fonseca et al., 2003]. As an example, if a feature class is named ‘Building’, it may actually designate only permanent buildings, or include precarious buildings, such as cabins, or huts. Moreover, it may even designate only habitations. Besides, a geographic database is produced at a specific scale of analysis and is therefore associated with a specific level of detail. Geographic features are then captured in the database consistently with its level of detail. For example, only buildings of area greater than $50m^2$ may be captured. Besides, in vector databases, the geometric representation of a given geographic feature may vary. As an example, a building may be represented by a polygon representing the surface covered by this building or by a point captured at the centre of the building. Since data capture for a given database is often done by several persons, homogeneity of data meaning within the database is ensured by storing all selection and representation criteria in specific textual documents, used as guideline for data capture, namely the database specifications (Fig. 1).

Such information about geodatasets capture process is a very rich and useful source of knowledge for geodata users: it helps them in understanding what kinds of real world geographical entities are represented in the dataset, with what feature types, what attributes, what geometry, and according to what selection and geometric modeling rules. The standard way of storing and managing this information consists in capturing it according to the ISO-19131 standard. However, this standard remains very informal regarding documentation of the data capture process itself, since this information can be represented only through free text into the standard. Therefore, it cannot be queried in an efficient way or compared from one geodataset to another in a way that helps users in understanding easily differences between datasets.



Building

Definition : Building of area greater than 20 m^2
Geometry : 3D polygon

Attributes :
- Identifier
- Data source
- Category
- Nature
- Height

Extensional definition: Look at values of attributes <category> and <nature>

Selection: All buildings of area greater than 50 m^2 are included.
Buildings of area lying between 20 m^2 and 50 m^2 are selected depending on their environment* and on their appearance**.
Buildings of area lower than 20 m^2 are not included. If they are very high, or if they are represented on the 1 :25000 scale map (e.g. monument, antenna, etc.), they are represented by an instance of the class "punctual building".

* Isolated small buildings which are 100m away from a dwelling house, and of area greater than 20 m^2 are included, whereas small buildings located in an urban area are not (e.g., small garage, etc.).
** Small buildings which look precarious (e.g. cabin, hut, etc.) are not included.

Geometry: Outer boundary of the building as it looks from above (most of the time, it is the roof boundary).
Inner courts, which are wider than 10 m, are represented by a hole in the surface representing the building.

Description	Real world and geometry	Geometry
Geometry of a house		

Several contiguous buildings, with the same <nature> and the same <category>, are considered as one single building (only the outer boundary is captured). Two contiguous buildings are represented separately in the cases when:
- the height difference between them is greater than 10 m (or 3 floors) ;
- each single building area is greater than 400 m^2 ;

Geometric constraint: Two contiguous buildings with different attributes values are represented by two surfaces with a common boundary.

Attribute: Category

Definition:	Type of a building according to its function and its appearance.
Type:	Enumerated.
Values:	Administrative / Industrial, agricultural, and commercial / Religious / Sports / Transports / Other.

Category = « Administrative »

Definition: Building with an administrative function.
Extensional definition: City hall | District police administration building| Sub-district police administration building

Category = « Industrial, agricultural or commercial »

Definition: Building with an industrial, agricultural or commercial function or typical appearance.
Extensional definition: slaughter-house| workshop | awning | electrical power plant | factory chimney (>20 m^2) | shopping center | stable | warehouse| industrial hangar | blast-furnace | hypermarket | department store| works| flour-milling works | car park | radar station | relay station | sawmill | glass-house | silo | factory

Fig. 1 Excerpt of the specification of the *BDTOPO®* database [IGN, 2002]

3 Related Works

Recent works in the field of geodata discovery mainly focused on solving semantic heterogeneity problems due to keyword-based search through catalogues. They are commonly based on approaches such as federated databases [Sheth and Larson, 1990] or mediators [Wiederhold, 1992] proposed in the field of computer sciences.

Paul and Ghosh [Paul and Ghosh, 2006] focus on integrating diverse spatial data repositories. Geodata discovery and retrieval is performed through a service-oriented architecture that uses a global ontology as information broker. To ensure semantic interoperability, application ontologies of data providers must be written using the shared vocabulary of the global ontology. In the GEON project, geodata source retrieval and integration is performed thanks to a semantic registration procedure [Nambiar et al., 2006]. Data providers are asked to describe mappings between their source schemas and one or more domain ontologies that are used to query all datasets in a uniform fashion. More generally, the issue of semantic annotation of geodata as the elicitation of relations between a data schema and a domain ontology by defining mappings between them is central in [Klien, 2008]. A rule-based approach combining semantic Web technologies and spatial analysis methods is introduced for automating this critical task. Rules are used to define conditions for identifying geospatial concepts. Spatial analysis procedures derived from these rules are used to determine whether a feature of a dataset represents an instance of a geospatial concept. Lassoued and colleagues [Lassoued et al., 2008] present an ontology-based mediation approach to perform geodata search across different OGC Catalogue Services for the Web (CSW). Ontologies are used as a mean to define semantics of metadata values used by different organisations, i.e. different CSW. Queries are written in the terms of a global ontology defining common metadata terms. This ontology is linked to local ontologies describing local metadata terms by an OWL mapping ontology. The mediator use these mappings to rewrite queries into local CSW ontologies vocabularies, send the requests to the local CSW, gather answers, translate them in the global ontology vocabulary and send results back to users. These works rather aim at enabling geodata discovery and retrieval. For example, if a user is looking for data that represent 'buildings', they aim at determining in which feature classes of the available datasets 'buildings' are represented, even if these classes have different names.

However, none of these works consider using knowledge from geographic database specifications in the geodata discovery process. Uitermark [Uitermark, 2001] proposes to use knowledge from geographic database specifications to build a federated schema and write mappings to link local schemas to the federated schema. Christensen [Christensen, 2006] presents a framework for developing production specifications of geodata. This is based on a built-in formal language named High Level Constraint Language (HLCL) based on first order predicate logic. Consistently with the works presented above, the author establishes a clear distinction between domain knowledge and its representation, i.e. between the real world entities and the database features. This distinction is materialised in the language's grammar through the use of different vocabularies for real world geographic entity types (referred to by "domain model") and database feature types (referred to by "conceptual domain"). Selection and representation rules are describes by mappings between domain model and conceptual domain. A close approach is proposed by Gesbert [Gesbert, 2005]. He also suggests a formal language for geographic database specifications formalisation. In this work, specifications are represented as expressive mappings between the database schema and a domain ontology. A comprehensive formal model enabling to formalise very complex data capture rules, such as road topological segmentation process, is developed in order to make all knowledge provided by geodata specifications available, either for users or for schema matching applications. These works aiming at geographic database specification formalisation globally rely on the same architectures as those dedicated to geodata discovery. Unfortunately languages proposed remain ad hoc solutions and must be adapted to become compliant with current standards recommended for geodata metadata management and semantic Web applications.

4 An Ontology-Based Model for Geodata Specifications Formalisation

4.1 Global Model: Two Levels of Ontologies

In this section, we describe general principles guiding our proposal for a formal model of geodata specifications. Following principles of the semantic Web, "the semantics of a given domain is usually encapsulated, elucidated and specified by an ontology [Kavouras and Kokla, 2008]", we thus propose to formalise each database specification by means of an application ontology, named "local specification ontology" (LSO). This ontology contains information such as selection criteria used to populate the database. For example, it formalises the fact that "only watercourses that are permanent and wider than 10 meters are represented in the feature class 'River' of the database", but also that "the geometry of features 'River' corresponds to the centreline of the modelled watercourses".

A key issue for successful use of formal specifications in an integration process is then to ensure enough homogeneity in the way they are formalised. This will be ensured by two means. The first one is to define unambiguously key concepts manipulated by the different local specification ontologies. In other words, we define a domain ontology, named "Specification Ontology" (SO), on which each LSO relies (taking the OWL vocabulary, we say that each LSO imports SO, cf. Fig. 2) [Abadie et al., 2010]. This ontology SO contains only concepts specific to geographic data specifications. It relies in turn on more general ontologies, for example for defining basic geometric types [Lieberman et al., 2007]. For example, this domain ontology SO formalises the concepts of data source and centreline, which are commonly used in many data specifications.

The second mean to ensure homogeneity between local specification ontologies is to define common rules to fill them. For example, we require that feature classes such as 'River' are modelled as "classes" in the OWL language, and that selection constraints are modelled as "axioms" including rules that restrict the possible

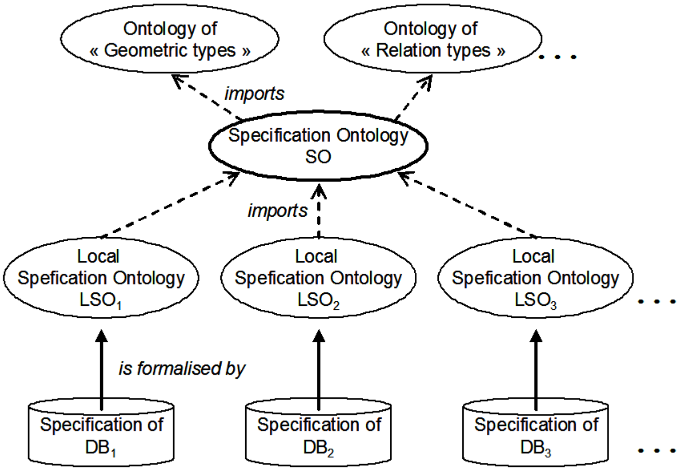

Ontology of « Geometric types »
Ontology of « Relation types »
. . .
imports
Specification Ontology SO
imports
Local Spefication Ontology LSO$_1$
Local Spefication Ontology LSO$_2$
Local Spefication Ontology LSO$_3$
. . .
is formalised by
Specification of DB$_1$
Specification of DB$_2$
Specification of DB$_3$
. . .

Fig. 2 Global framework for formalising database specifications

interpretations for the defined term, those axioms being defined by means of concepts and relations defined in SO and LSO (see section 4.3 for details).

4.2 The Specification Ontology (SO)

This section details the Specification Ontology (SO), which is considered here as a common semantic model represented in OWL and providing a unified view of the formal geographic database specification concepts to be shared. The elements (concepts and properties) of the SO ontology are referred to when building Local Specification Ontologies (LSO), as explained more in details in the next section. The structure of the SO ontology is depicted in Fig. 3 and described below.

The SO ontology consists of a set of classes that are related either by a subsumption relationship 'is-a' or by other relations we introduced. As reusing existing domain ontologies is recommended in order to enable interoperability between computing applications [Simperl, 2009] some of its classes and relations are imported from existing ontologies adopted in the geographic domain. The main classes and relations of the SO ontology are summarized in the following subsections.

The 'Feature' and 'GeographicEntity' Classes

The key idea guiding our model is that:

"semantics provides meaning by associating the representing to what is represented in the real world [Kavouras and Kokla, 2008]". A clear distinction between concepts manipulated in the data schema and concepts describing the real world must be made, even if textual specifications may use the same words to designate them.

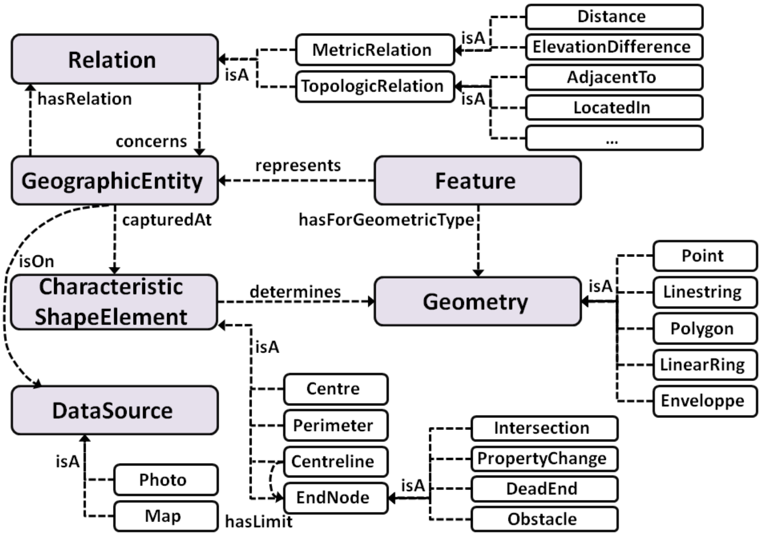


Fig. 3 General Structure of the Specification Ontology (SO)

Let us take a simple example to illustrate that. If a specification states that the feature class ‘Buildings’ is defined as “Buildings bigger than $50m^2$”, it is clear that the first ‘Building’ refers to the name of the feature class representing only some particular buildings in a particular database, while the term ‘Building’ in the definition refers to a general and shared concept of the real word. Those two buildings have very different meaning that must be separated for an efficient formalisation of semantics. The SO ontology thus includes two base classes which are: ‘Feature’ and ‘GeographicEntity’ (Fig. 3). The class ‘Feature’ is imported from GeoRSS-Simple[1], a simple serialization of the GeoRSS feature model [Lieberman et al., 2007] following the general principles of the ISO General Feature Model, and recommended by the W3C Geospatial Ontologies incubator group for the description of geospatial resources on the Web. This ontology is used, in a first approach, as a simple way to model the schema classes of a particular geographic database. The class ‘GeographicEntity’ models the real world entities and their classical properties like their height, width, length, surface, perimeter, etc. Real world geographic entity properties are represented in OWL by DatatypeProperties with ‘GeographicEntity’ as a domain and OWL data type $xsd : double$ as a range. The relation between the class ‘Feature’ and the class ‘GeographicEntity’ is materialized by the relationship *represents*, which is an OWL ObjectProperty with ‘Feature’ as a domain and ‘GeographicEntity’ as a range. In OWL, this relation is expressed with the following axiom associated with the ‘Feature’ class[2]:

$$\texttt{gml:Feature} \sqsubseteq \exists \texttt{so:represents.so:GeographicEntity}$$

[1] http://mapbureau.com/neogeo/neogeo.owl

[2] Axioms are represented with the Description Logics Syntax underlying the OWL language.

The Relation Class

Other relations, which are OWL ObjectProperties, are used to connect the class 'GeographicEntity' to the other classes in the ontology SO. OWL ObjectProperties are binary predicates; they relate two classes or two instances. However, we need to use also ternary relations, such as distance, which relates three elements: two geographic entities and a restriction on a DatatypeProperty. In order to allow ternary relations, and in general n-ary relations, to be represented in OWL, we used the reification in its traditional form[3]. The reification consists in introducing a new class (for example Distance on Fig.3) with two properties: a DatatypeProperty called value specifying the value of the distance, and an ObjectProperty concerns referring to a geographic entity. Then, we obtain a binary relation between a geographic entity and the reified class Distance.

Topological relations, like the relation *locatedIn*, can be represented as OWL ObjectProperties, since they are binary relations. This is the case in the spatial relations ontology implemented by the Ordnance Survey[4]. In our SO ontology, we propose to use both solutions: on the one hand, topological relations can be modeled as binary relations through object properties, and on the other hand, the same relations can be modeled as n-ary relations. The main interest of this double formalisation is that reification represents relations as classes associated with a rich semantics. Then it makes it possible to explain exactly what we mean by each relation, especially in the context of spatial relations which are very complex [Hudelot et al., 2008].

The Geometry and GeometricModeling Classes

Every instance of a geographic vector database feature class has a geometrical representation, which describes the location and the shape of its corresponding real world entity, consistently with the database level of detail. Therefore, a specification also details, for each feature class, how instances geometry shall be captured.

The Geometry class is imported from the GeoRSS-Simple ontology. It models the different types of geometries used for geographic data, like Polygon. The Geometry class is also related to the 'Feature' class with the *hasForGeometry* OWL ObjectProperty:

$$\texttt{gml:Feature} \sqsubseteq \exists \texttt{so:hasForGeometry.gml:Geometry}$$

The *CharacteristicShapeElement* class models the characteristic shape elements of geographic entities, which shall be captured to instantiate database feature class instances geometry, like *Centre*, *Centreline* or *Perimeter*. For example, an instance of a feature class 'Building' can be captured by drawing the perimeter of the corresponding real world 'building' in order to be stored in the database as a polygon. The *CharacteristicShapeElement* class is related to the class *Geometry* with the OWL ObjectProperty determines; each sub-class of *CharacteristicShapeElement* has an

[3] `http://www.w3.org/TR/swbp-n-aryRelations/`

[4] `http://www.ordnancesurvey.co.uk/ontology/SpatialRelations.owl`

axiom specifying the type of geometry it determines. For example, the class *Perimeter* is defined as follows:

$$\texttt{so:Perimeter} \sqsubseteq \exists \texttt{so:determines.gml:Polygon}$$

The DataSource Class

The *DataSource* class models the different digital or paper supports on which a geographic entity can be visible or present. Now, it mainly includes two sub-classes, (but new classes could be added): *Photograph* and *Map*, both associated with an OWL DatatypeProperty specifying their scale. These classes serve to express specifications of the form "... present on the map", or "... visible on the photo".

$$\texttt{so:DataSource} \sqsubseteq \exists \texttt{so:scale.string}$$

4.3 Local Ontologies Implementing Rules

For each database to be integrated, a local specification ontology (LSO) is created. It describes the database schema and the rules used to populate its feature classes by capturing geographic entities. This local specification ontology imports the specification ontology (SO), described in section 4.2, and uses it to formalise the specification of that specific database as shown on figure 4.

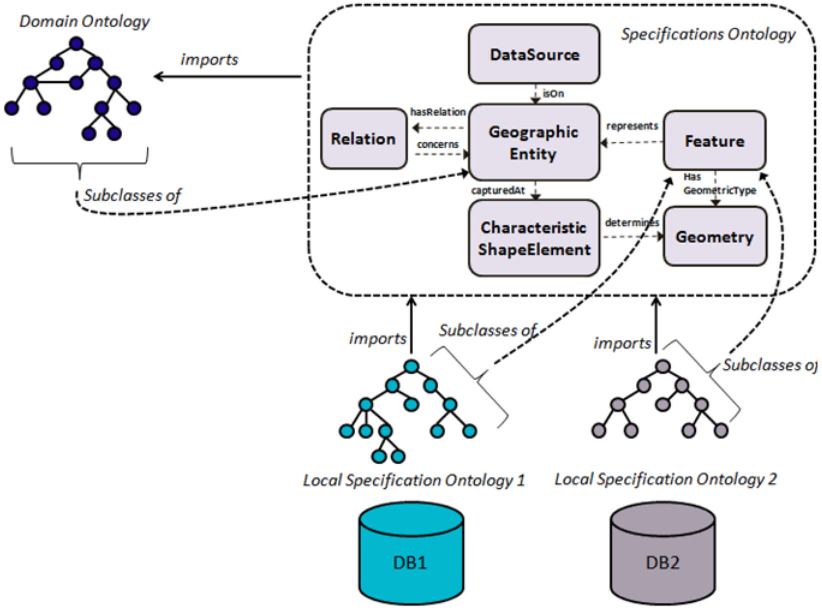


Fig. 4 Local specification ontologies

How to Represent Database Schema Entities?

A first step to formalise a specification consists in translating the database schema into OWL formalism. This is done according to a fairly intuitive strategy already presented in [Abadie, 2009]. Each schema class represents an object-oriented abstraction of real world entities. Therefore, in our local specification ontology, each feature class will be translated into an OWL class. As they represent schema feature classes, these OWL classes will be created as sub-classes of the SO class ‘Feature’. OWL class datatype properties, object properties and *subClassOf* relations are straightly derived from their respective feature class attributes, associations, and inheritance relations.

Moreover, it is a common modeling practice for geographic databases to simplify the schema structure by merging semantically close feature classes into a single class. In such cases, the specific nature of each instance of the feature class is defined more accurately by an attribute (usually named ‘nature’ or ‘type’). Most of the time, these attribute values are terms that designate geographic concepts. As an example, a feature class ‘Water Point’ has an attribute ‘nature’ with possible values ‘cistern’, ‘fountain’, ‘spring’ or ‘well’. Besides, it happens frequently that instances of such feature classes, having different natures, have different specifications, e.g. different selection criteria. We propose to translate the values of these specific attributes into OWL classes, subsumed by the OWL class derived from their respective feature class in the database schema, in order to make their specification formalisation easier.

We have implemented a generic translator, developed with the protégé-owl API[5]. It takes an ISO 19109 [ISO-TC-2011, 2001] schema as input and converts it into OWL ontology elements [W3C, 2004], according to the strategy presented above. Figure 5 shows how a piece of *BDCARTO*® [IGN, 2005] schema is translated into OWL format.

How to Represent Real World Geographic Entities?

As specifications detail how real world geographic entities are captured in a given database, we need to represent these geographic entities in our local specification ontology, which consists in making explicit what Partridge [Partridge, 2002] calls the domain ontology which underlies the database. The geographic entities of this underlying domain ontology can be derived from the specification text. Actually, a specification describes the database structure and content by using geographic vocabulary. An intuitive method to build this domain ontology consists in retrieving in the specification text the specific terms used to designate real world geographic entities and to use them as labels for our OWL classes. These OWL classes are represented in our local specification ontology as sub-classes of ‘GeographicEntity’.

[5] http://protege.stanford.edu/plugins/owl/api/

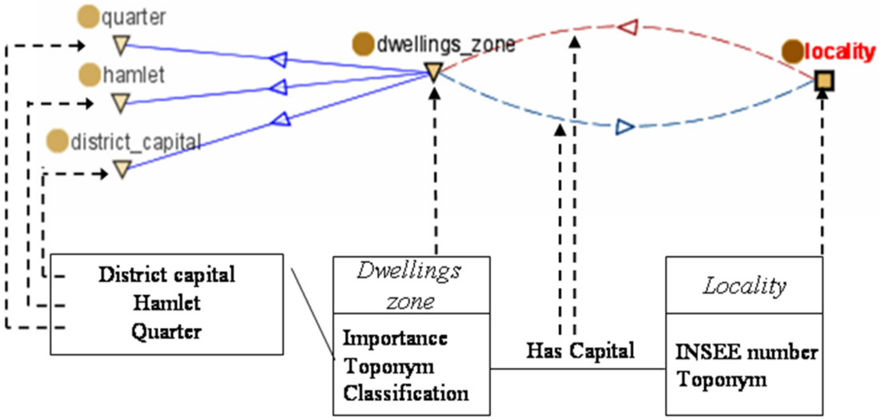


Fig. 5 Translating *BDCARTO*® schema (at the bottom of the image) into OWL ontology (piece of ontology visualized with Protégé, at the top of the image)

Building such a domain ontology can be done in a more or less structured manner. In order to go a step further, relations between concepts, like subsumption or meronymy relations, can be created by analysing the linguistic relationships between geographic terms provided by the specification text. This can be done semi-automatically thanks to natural language processing tools, such as the one proposed in [Aussenac-Gilles and Kamel, 2009]. Finally, this domain ontology can be improved by a manual analysis. This is a longer and harder task but provides a semantically richer ontology and therefore better integration results.

For our study, we use a bilingual (French/English) taxonomy built at the COGIT laboratory from a semi-automated analysis of geographic terms encountered in several data specifications [Abadie and Mustière, 2010] (Fig. 4). This taxonomy of the topographic domain contains more than 760 concepts. In the future, we aim at using a richer ontology, in terms of number of concepts and in terms of description of those concepts. The production of this ontology is one of the goals of the undergoing GéOnto project [Mustière et al., 2009]. The enrichment of the original taxonomy is made thanks to the analysis of two types of textual documents: technical specifications of geographic databases and travelogues. It is based on automated language processing techniques, ontology alignment and also on external knowledge sources like dictionaries and gazetteers of place names.

How to Link Features to Geographic Entities?

As explained in section 4.3, the relationship between database features and geographic entities is formalised thanks to an ObjectProperty named *represents*. It enables us to instantiate several kinds of specification rules. As an example, selection constraints, which specify, for each feature class, the nature of real world entities that must be represented are formalised by *someValuesFrom* restrictions[6]:

$$\texttt{lso:Spring} \equiv \exists\texttt{so:represents.(topo:Spring}\sqcup \\ \texttt{topo:Resurgence}\sqcup\texttt{topo:Outcropping)}$$

Moreover, a geographic database is associated with a given level of detail. Therefore, real world entities must be captured consistently with this level of detail. Thus specifications express geometric constraints on the size of geographic entities that shall be captured in a specific feature class. As an example, a specification precises that 'basins' are captured in the database class 'Water Body' if their length is greater than 10m. In OWL, the range of a *DatatypeProperty* is a simple built-in data type like a double, integer or string. However, in this case, we need to make a different cardinality restriction. This becomes possible with the new version of OWL, namely OWL 2[7] which is based on a more expressive Description Logic *(SROIQ*) and which provides some new interesting features[8], one of which is the data type restriction construct, which allows new data types to be defined by restricting the built-in data types in various ways [Grau et al., 2008].The constraints on the size of 'basins' presented above can then be formalised as follows:

$$\texttt{lso:WaterBody} \equiv \exists\texttt{so:represents.(topo:Basin}\sqcap \\ \texttt{so:length some double[>10.0])}$$

Contextual constraints state that real world entities are captured in the database if they are really significant in the landscape depending on their environment, or if they are mentioned on a reference data source. In the former case, the constraints used will deal with real world geographic entities relationships. As an example, a specification of the feature class 'GuardedShelter' states that: "... 'mountain hotels' which are located in a 'National Park' or in its vicinity (less than 2km away from the park) are also included". These kinds of constraints are formalised with restrictions on metric relations and topologic relations defined in the specifications ontology:

$$\texttt{lso:GuardedShelter} \equiv \\ \exists\texttt{so:represents.((topo:Mountain\_Hotel}\sqcap \\ \exists\texttt{so:locatedIn.topo:National\_Park)}\sqcup\texttt{(topo:Mountain\_Hotel}\sqcap \\ \exists\texttt{so:hasRelation.(so:Distance}\sqcap \\ \exists\texttt{so:concerns. topo:National\_Park}\sqcap \\ \texttt{so:value some double[<2000.0])))}$$

In the later case, when geographic entities are required to be mentioned on a specific data source, this constraint can be formalised thanks to a someValueFrom restriction: instances of a given geographic entity are related to instances of *DataSource*

[6] In all examples given here, entities preceded by the namespace *so* : belong to the SO model, those preceded by *lso* : belong to local specification ontologies, and those preceded by *topo* : belong to the topographic domain ontology.

[7] http://www.w3.org/TR/owl2-overview/

[8] http://www.w3.org/TR/2009/REC-owl2-new-features-20091027/

via the *isOn* ObjectProperty. For example, the ‘Water Point’ feature class specification, which precises that “ ‘Springs’ are represented if they are mentioned on the 1:25000 map”, will be translated into:

$$\begin{gathered}\texttt{lso:WaterPoint} \equiv \exists\texttt{so:represents.(topo:Spring}\sqcap\\ \exists\texttt{so:isOn (so:Map}\sqcap\texttt{so:scale value 1/25000))}\end{gathered}$$

Besides, specifications define constraints on specific real world entity properties, such as “only outdoor ‘swimming-pools’ are captured”. Geographic entity properties are defined in our model by DatatypeProperties and constraints on their values are formalised thanks to restrictions on these properties:

$$\begin{gathered}\texttt{lso:SwimmingPool} \equiv \exists\texttt{so:represents.(topo:Swimming\_pool} \sqcap\\ \texttt{lso:isOutdoor value true)}\end{gathered}$$

How to Formalise Geometry Instantiation Rules?

Geometry instantiation rules, such as “Watercourse segment geometry is represented by a line drawn along the ‘river’ centreline”, define what geometric type shall be used for a feature class, and what characteristic shape elements of real world entities shall be depicted. Both aspects are taken into account in our SO ontology, so that the geometry instantiation rule presented above can be formalised with two someValueFrom restrictions:

$$\begin{gathered}\texttt{lso:db\_WatercourseSegment} \equiv\\ \exists\texttt{so:hasForGeometry.gml:LineString}\sqcap\\ \texttt{so:represents.(lso:River}\sqcap\exists\texttt{so:capturedAt.so:CentreLine)}\end{gathered}$$

How to Formalise Attribute Instantiation Rules?

A feature class specification also defines the meaning of class attributes and explains how their values shall be filled. However, by attribute instantiation rules, we do not mean cardinality or data type constraints, but rather rules which define precisely how attribute values shall be determined for each feature class instance, like: “The attribute ‘width’ of the feature class ‘Hydrographic segment’ takes the value ‘small’ if the width of this ‘river section’ lies between 0 and 10 meters”. Such rules can typically not be directly represented in OWL since they are constraints between fillers of two different properties. As a consequence, we propose to use the Semantic Web Rule Language (SWRL)[9] rules to formalise them. As a matter of fact, SWRL was designed to add additional expressivity to OWL. The specification rule presented above will be formalised as follows:$\texttt{topo:river(?x)}\wedge\texttt{so:width(?x,?y)}\wedge$ $\texttt{swrlb:GreatherThan(?y,0)}$
$\wedge\ \texttt{swrlb:LesserThan(?y,10)}\wedge\texttt{lso:HydrographicSegment(?z)}\wedge$
$\texttt{so:represents(?z,?x)} \Rightarrow \texttt{lso:size(?z, ``not wide'')}$

[9] http://www.w3.org/Submission/SWRL/

5 Implementation of the Proposed Model: A User-Friendly Web Interface for Geospatial Data Discovery

The proposed model was applied in the framework for geographic databases content discovery. The goal is to enable users to discover which entities of interest are represented in a given database, and how they are represented. The purpose of this system is to provide, through a user-friendly interface, complex information on geographic data, previously not accessible or only accessible by reading the complex specification files. More precisely, the goals of our system are the followings:

- Guiding the user in specifying which information is of interest for her/him with terms from the domain ontology (e.g. river), rather than technical terms used in the database schema (e.g. 'hydrographic section' or even 'hydr_lin');
- Retrieving in the database data corresponding to the user's need;
- Providing additional information about the data: which real world entities are represented (e.g. all the watercourses or only those with permanent flow); how they are represented in the database (in which class, with which attribute values?); and how are they distinguished from other entities (e.g. does the information in the database allow to make the distinction between man-made canal and natural watercourses?);
- Visualizing the data corresponding to the user's need using web mapping techniques.

5.1 Architecture

Fig. 6 shows the global architecture of our system. As explained before, our system allows a user to better understand geographic databases content thanks to the use of ontologies that formalise, as explained before, the specifications associated with these databases. It also allows a user to visually compare geographic datasets throw a web mapping solution allowing the visualization of the data. In this system, the user expresses his/her query using terms from the domain topographic ontology. This way, s/he is able to know about several geographic databases contents since they are described using the same vocabulary.

Our system runs on a client-server architecture, including a web mapping solution for data visualization in the client's side. It is composed of three important modules.

1. The search module: this module guides the user, through an auto-completion solution, to express his/her query, i.e. to specify which data s/he is looking for using terms designating concepts in the topographic domain ontology. Indeed, the interest of the topographic domain ontology is two-fold: it is supposed to contain a shared vocabulary rather than a technical one, and all formalised specifications of databases rely on it. The topographic domain ontology is thus used to express user's needs and is a pivot in our system;

2. The information extraction module: once the user selects the label of a given geographic concept of interest, the system extracts, from the local specification ontologies, information about the data referred to by this term. This piece of information, including definitions, geometries of represented objects etc., is sent to the user;
3. The cartographic module: the system uses information obtained from the local specification ontologies to retrieve data corresponding to the user's need in the different geographic databases. Data are sent to the user for visualization through a web mapping solution.

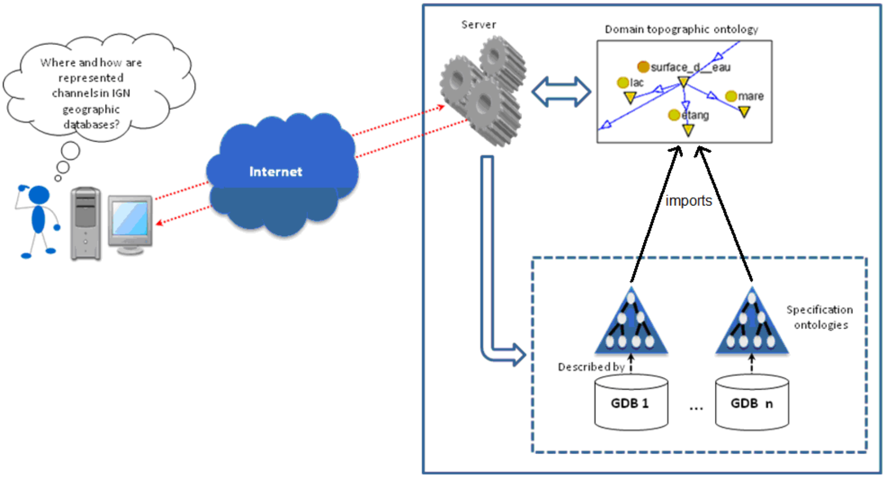


Fig. 6 System architecture

Illustrating Example

Let us consider the following example, and suppose that the user wants to know where and how are represented resurgences in the databases annotated with this ontology.

$$\texttt{lso:Spring} \equiv \exists\texttt{"so:represents.(topo:Spring} \sqcup \texttt{topo:Resurgence} \sqcup \texttt{topo:Outcropping)}$$

From the free text 'resurgences' filled by the user, the system proposes him/her the term 'Resurgence' included in the topographic domain ontology. Next, the system retrieves all classes of the local specification ontologies which are annotated with the concept of resurgence, such as the class 'Spring', among others. After that, the system retrieves data from the table 'Spring' of the database, where resurgences are stored, and sends them to the user as well as other information from the local specification ontology such as the type of geometry of the data, and the capture constraints on resurgences. Finally, a web mapping solution allows the user to visualize capture constraints and available data themselves, in order to assess whether they fit to his/her needs. This may be simultaneously done for several datasets, and thus enables a precise comparison of specific advantages and drawbacks of each dataset.

5.2 *Interface*

The system interface is composed of three blocks (Fig. 7): the first block allows the user to enter his query; the second block extracts information from local specification ontologies that correspond to the user's need; the third block displays data samples searched by the user. The interface of the system is conceived so that the user can visualize both data samples corresponding to his/her need and information describing them.

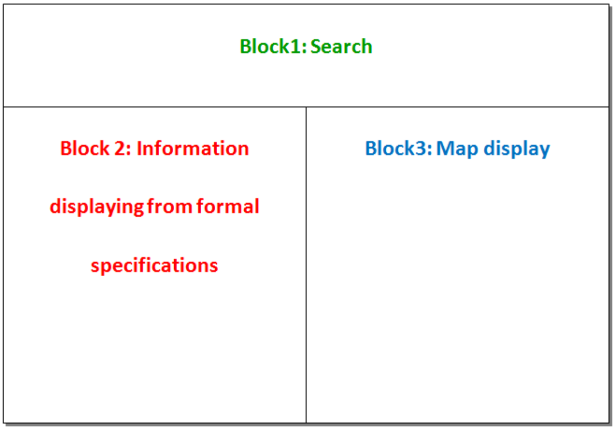


Fig. 7 System Interface

5.3 *Implementation*

The proposed system is a web application allowing serving a large number of users (Fig. 8). A prototype has been implemented using two local specification ontologies associated with two IGN databases (*BDTOPO*® and *BDCARTO*®). These ontologies are represented in OWL 2 and are actually restricted to the hydrographic theme. Programs running on the server are implemented in Java, using the OWL 2 API[10] for parsing ontologies. For the web side, JSP, HTML and JavaScript languages and JQuery library are used. The web server used is Apache Tomcat, since it can interpret JSP pages. The cartographic server used by the system is Geoserver[11], since it is developed as a J2EE application and can be executed on a Tomcat server. This way,

[10] http://owlapi.sourceforge.net/

[11] http://geoserver.org/display/GEOS/Welcome

the system requires only one server. Postgres with the Postgis extension is used as database management system. Two geographic datasets were used, subsets of data from the *BDTOPO*® and *BDCARTO*® products. The system uses also the WMS protocol for data displaying and the Géoportail API in its cartographic part, in order to access to IGN base maps.

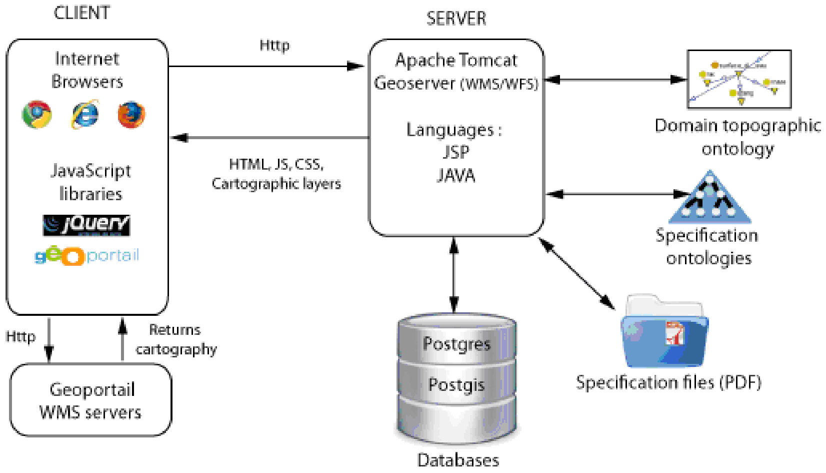
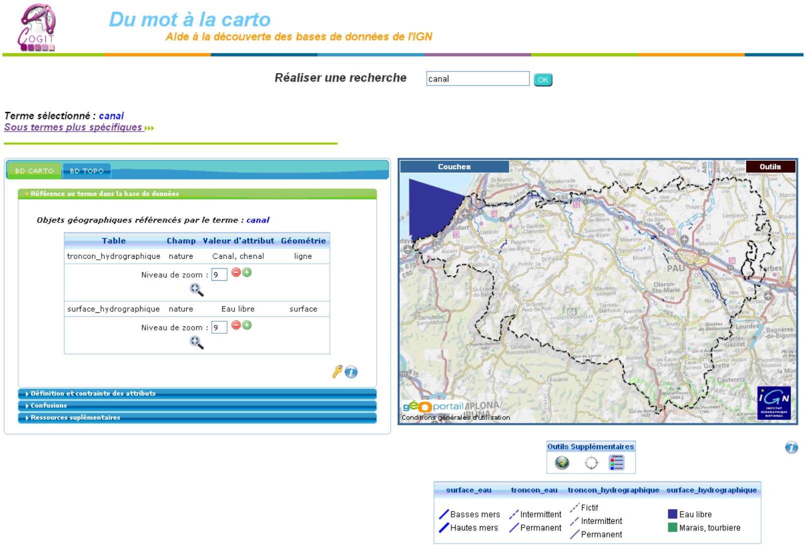


Fig. 8 System implementation

The web interface of the implemented system is shown on Fig. 9. It is composed of three parts: the first part consists in a text-field for entering the query as in a classical search engine where the user can specify which data she/he is interested in (here 'channels'). The second part (on the left) consists in a set of tabs; each tab corresponds to a database and provides information about data interesting the user in this database. The third part is the cartographic visualization of the data corresponding to the user's need in each database. When the user switches from one tab to another one, the cartographic visualization of the corresponding data is automatically updated.

The search module was implemented using the auto-completion script available in the JQuery library, which works asynchronously and displays results as a list of terms. In Protégé software –ontology editor–, concepts cannot contain spaces or quotes; they are generally replaced with underscores. However, in our topographic domain ontology, concepts are associated with labels, represented as OWL annotations, both in English and French. So, for ergonomic reasons, in our auto-completion procedure we display these labels instead of the concept names themselves.

In order to take into account typos when the user enters his/her query, the system computes a distance between the typed string and the concepts names in the domain topographic ontology, using the Levenshtein edit distance [Levenshtein, 1966] which was normalized in our system using the method proposed by Yujian and


COGIT
Couches
Outils
géoportail
Conditions générales d'utilisation
IGN INSTITUT GÉOGRAPHIQUE NATIONAL

Du mot à la carto

Aide à la découverte des bases de données de l'IGN

Réaliser une recherche canal OK

Terme sélectionné : canal

Sous termes plus spécifiques

BD CARTO | BD TOPO

Référence au terme dans la base de données

Objets géographiques référencés par le terme : canal

Table	Champ	Valeur d'attribut	Géométrie
troncon_hydrographique	nature	Canal, chenal	ligne
Niveau de zoom : 9			
surface_hydrographique	nature	Eau libre	surface
Niveau de zoom : 9			

Définition et contrainte des attributs

Confusions

Ressources suplémentaires

Outils Supplémentaires

surface_eau	troncon_eau	troncon_hydrographique	surface_hydrographique
Basses mers Hautes mers	Intermittent Permanent	Fictif Intermittent Permanent	🟦 Eau libre 🟩 Marais, tourbiere

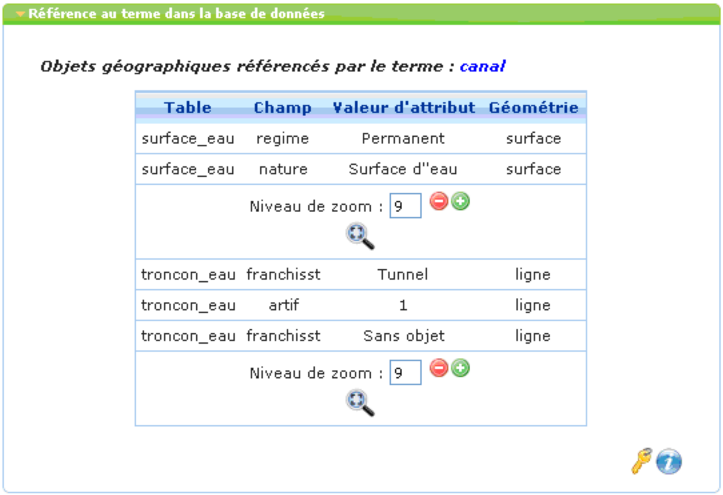
Fig. 9 The web interface of the implemented prototype

Bo [Yujian and Bo, 2007]. This way, the closest concept labels will be proposed to the user. For example, if the user types ‘canel’ instead of ‘canal’ (which means channel in French), then the system will still find ‘canal’. The system also allows the user to refine her/his query by proposing to her/him terms designating concepts that are semantically close[12] to the concept ‘canal’ in the topographic ontology, and for which data exists.

The information extraction module extracts from the local specification ontologies, information about the data referred to by the term specified by the user. In the case of ‘canal’ the system will extract information about precise localizations of channels in the databases and other information like the type of geometry of channels in the database (polygons or polylines). All those pieces of information are sent to the user and displayed as shown on Fig. 10. For each considered database, information display is organized in an accordion composed of four sections.

The first section (Fig. 10) indicates which features of the database refer to the term selected by the user: it describes how these features are represented in one or several classes, and with which attribute values they are modeled. Here, channels are described in two different classes ‘water_surface’ and ‘watercourse segment’, that may be shown to the user, and objects representing channels in the class ‘watercourse segment’ have in particular the specific attribute value ‘artificial = true’, which allows to distinguish them from natural watercourses.

[12] Here, by semantically close we mean concepts that are at a distance, in terms of the number of edges, of one or two from the concept ‘channel’ in the domain ontology.



▾ Référence au terme dans la base de données

Objets géographiques référencés par le terme : canal

Table	Champ	Valeur d'attribut	Géométrie
surface_eau	regime	Permanent	surface
surface_eau	nature	Surface d"eau	surface
		Niveau de zoom : 9	
troncon_eau	franchisst	Tunnel	ligne
troncon_eau	artif	1	ligne
troncon_eau	franchisst	Sans objet	ligne
		Niveau de zoom : 9	

Fig. 10 Information on data localization in the database

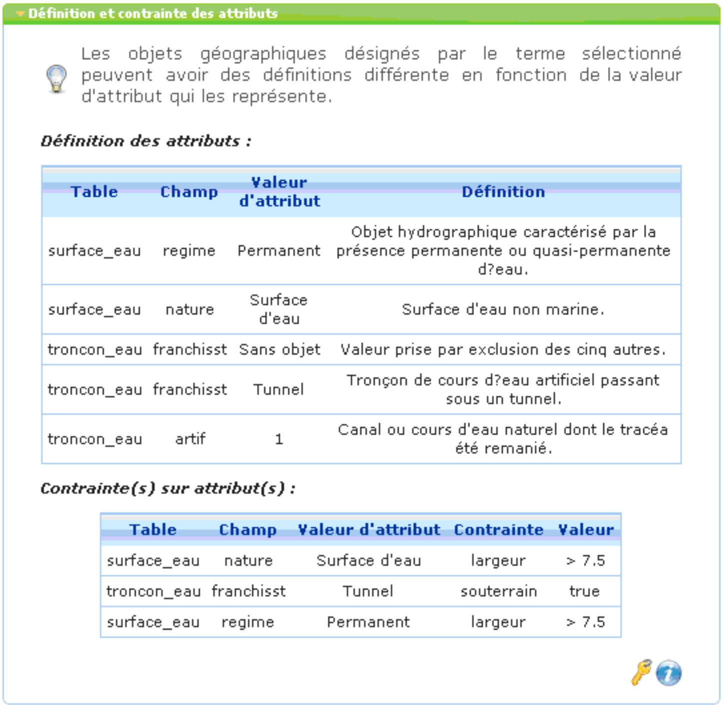
The second section presents more in detail specifications related to the considered features (Fig. 11); A first part details definitions of attribute values used to model the specific concept. A second part shows the constraints that should be satisfied by geographic entities in order to be included in the database (here, channels are represented by surfaces only if they are wider than 7.5 m).

The third section (Fig. 12) lists, without distinction, all real world entities which are represented in a given table with the same attribute values. For example, 'water-course segments' with field 'artificial=true' may represent either channels, reaches and watercourses ('canal', 'bief' and 'cours d'eau' in French), without any possibility to distinguish between them. This is a way to explain the granularity of attributes describing the nature of objects, which may be of importance for the user.

The fourth section provides additional available information. The user can download original textual specification files corresponding to his/her query, and is able to view a subset of the database, and download it in KML format in order to be able to use it on any GIS software.

The cartographic module is implemented using to the Géoportail API, and the displayed layers are controlled by the system according to the term selected by the user. In order to display only geographic objects corresponding to the user's need, the system filters the WMS layers served by Geoserver, using CQL queries which allow the creation of layers with the Géoportail API. The system displays automatically the layers corresponding to a specific database when the user switches from one tab into another one.

In addition to functions offered by the GéÃ©oportail API, new ones were implemented in our system, like retrieving a place using its address, adding vector layers or images in order to allow the user to compare his/her data with those proposed by our system, etc.



Les objets géographiques désignés par le terme sélectionné peuvent avoir des définitions différente en fonction de la valeur d'attribut qui les représente.

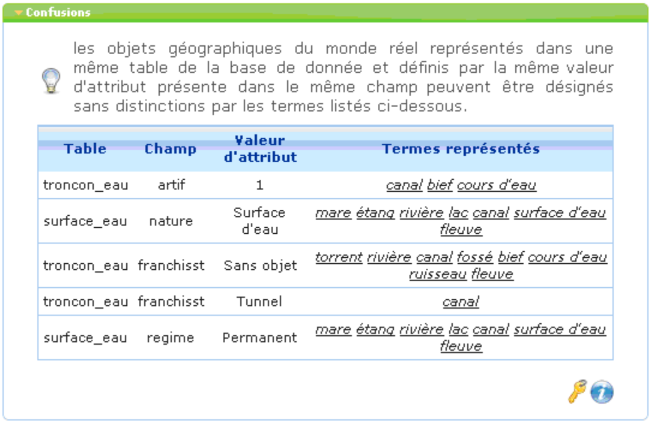
Définition des attributs :

Table	Champ	Valeur d'attribut	Définition
surface_eau	regime	Permanent	Objet hydrographique caractérisé par la présence permanente ou quasi-permanente d?eau.
surface_eau	nature	Surface d'eau	Surface d'eau non marine.
troncon_eau	franchisst	Sans objet	Valeur prise par exclusion des cinq autres.
troncon_eau	franchisst	Tunnel	Tronçon de cours d?eau artificiel passant sous un tunnel.
troncon_eau	artif	1	Canal ou cours d'eau naturel dont le tracéa été remanié.

Contrainte(s) sur attribut(s) :

Table	Champ	Valeur d'attribut	Contrainte	Valeur
surface_eau	nature	Surface d'eau	largeur	> 7.5
troncon_eau	franchisst	Tunnel	souterrain	true
surface_eau	regime	Permanent	largeur	> 7.5

Fig. 11 Definitions and constraints on data included in the database



les objets géographiques du monde réel représentés dans une même table de la base de donnée et définis par la même valeur d'attribut présente dans le même champ peuvent être désignés sans distinctions par les termes listés ci-dessous.

Table	Champ	Valeur d'attribut	Termes représentés
troncon_eau	artif	1	canal bief cours d'eau
surface_eau	nature	Surface d'eau	mare étang rivière lac canal surface d'eau fleuve
troncon_eau	franchisst	Sans objet	torrent rivière canal fossé bief cours d'eau ruisseau fleuve
troncon_eau	franchisst	Tunnel	canal
surface_eau	regime	Permanent	mare étang rivière lac canal surface d'eau fleuve

Fig. 12 Confusions

Thanks to our system, it is now possible to compare different databases contents with regards to a specific user's need. The implemented prototype allows comparing the *BDTOPO*® and *BDCARTO*® contents. For example, if the user wants to know which databases better represent channels, s/he can quickly have detailed

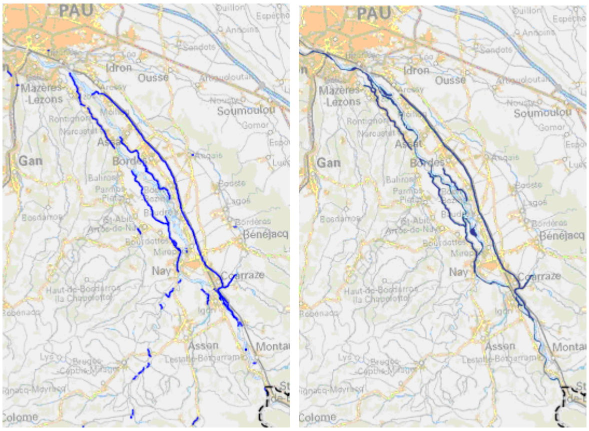


Fig. 13 (left) Channel representation in BDTOPO, (right) Channel representation in BD-CARTO

information about channels in each database and visualize channels in both databases as shown on Fig. 13.

6 Conclusion and Perspectives

In this paper we proposed an OWL2-based model for geographic database specification formalisation, which aims at eliciting geographic databases semantics by describing the link between data and what they represent. Two levels of formalisation are distinguished: the key concepts used in data specifications are specified in a specification domain ontology (SO), whereas knowledge contained in one given database specification is described in a specification application ontology (LSO) which uses the concepts of the specification ontology. The model for the formalisation of geographic database specifications that we propose has been implemented on the specifications of different databases in order to check whether it really enables to represent most of specifications contents. However, even if this model proved to be generic enough to represent different specifications, there remain specifications rules that are not formalised now in our proposal, such as fuzzy specifications rules. This is due to the fact that actual standard OWL version does not handle uncertainty. Second, vague rules that must be formalised can usually have many subjective interpretations: “ ‘Basins’, ‘Wells’ and ‘Wash-houses’ are captured if they are exceptional”. As a consequence, we propose to keep such vague rules as annotations in natural language (represented by OWL annotation properties).

Once the specification of each database that we want to integrate has been formalised in a LSO ontology, we can compare these LSO ontologies to automatically derive mappings between database schemas. For that purpose, a tool is being implemented in Java with the OWL API. It takes two LSO ontologies as input and outputs expressive mappings based on the Geo Ontology Mapping Language [Reitz et al., 2009] defined in the HUMBOLDT project as a geographic databases specific extension of the Ontology Mapping Language [Euzenat et al., 2007]. The use of a reasoner (Hermit[13]) enables us first to check the consistency of each formal specification. Moreover, when both LSO ontologies are merged, it can infer *equivalentClass* and *subClassOf* relations between feature classes of our databases schemas. However, most of the time, relations between geographic databases feature classes are not direct equivalence or subsumption relations, but rather equivalence relations between instances of subsets of each databases feature class, which represents the same geographic entities. In order to find such relations, an application is being implemented for comparing axioms of both LSO ontologies, and derive expressive schema mappings from comparison results.

The proposed model was used in a system for geographic databases content discovery. This system allows a better comprehension of the geographic databases content, thanks to the formalization and an appropriate display of their specifications coupled with a web mapping solution allowing a visual comparison of the data of interest. The implemented prototype of the system using data from the IGN shows the feasibility and usefulness of the approach for geodata understanding and comparison.

In the future, we plan to improve the system with respect to several aspects. For expressing his/her need, the user has actually the possibility to select one concept name from the topographic domain ontology. It would be interesting to allow her/him to select more than one concept name in order to be able to compare data representing different geographic entities. For example, allowing comparing data which represent both channels and roads. Moreover, the auto-completion method could be improved, for example by returning to the user concept names in a form respecting the taxonomic structure of the topographic domain ontology. This would allow the user to better specify which data s/he is interested in. The implemented prototype is actually restricted to the hydrographic theme of two geographic databases from IGN. It would be interesting to extend it to other geographic databases from IGN and outside IGN in order to allow users to compare databases issued from more different conceptualizations and points of view. Moreover, we plan to associate to each term of the ontology its meaning as a textual definition (annotation) which will be displayed to the user when formulating her query.

Regarding the ontology, we plan to enrich it by using existing techniques such as the one proposed in [Blessing and Schütze, 2010], which extracts geospatial relations in text.

Acknowledgements. This work is partly founded by the French Research Agency through the GeOnto project ANR-O7-MDCO-005 on "creation, alignment, comparison and use of geographic ontologies" (http://geonto.lri.fr/).

[13] http://hermit-reasoner.com/

References

[Abadie, 2009] Abadie, N.: Schema matching based on attribute values and background ontology. In: 12th AGILE International Conference on Geographic Information Science, Hannover, Germany (2009)

[Abadie et al., 2010] Abadie, N., Mechouche, A., Mustiére, S.: Poster: Owl-based formalisation of geographic databases specifications. In: International Conference on Knowledge Engineering and Knowledge Management, EKAW (2010)

[Abadie and Mustière, 2010] Abadie, N., Mustière, S.: Constitution et exploitation d'une taxonomie géographique à partir des spécifications de bases de données. Revue Internationale de Géomatique 20(2), 145–174 (2010)

[Aussenac-Gilles and Kamel, 2009] Aussenac-Gilles, N., Kamel, M.: Ontology learning by analyzing XML document structure and content. In: KEOD, pp. 159–165 (2009)

[Blessing and Schütze, 2010] Blessing, A., Schütze, H.: Self-annotation for fine-grained geospatial relation extraction. In: COLING, pp. 80–88 (2010)

[Christensen, 2006] Christensen, J.V.: Formalizing specifications for geographic information. In: Proceedings of the 9th AGILE Conference on Geographic Information Science, pp. 186–194 (2006)

[Euzenat et al., 2007] Euzenat, J., Scharffe, F., Zimmermann, A.: Expressive alignment language and implementation. deliverable. Knowledge Web NoE (2007)

[Fonseca et al., 2003] Fonseca, F., Davis, C., Câmara, G.: Bridging ontologies and conceptual schemas in geographic information integration. Geoinformatica 7, 355–378 (2003)

[Gesbert, 2005] Gesbert, N.: Etude de la formalisation des spécifications de bases de données géographiques en vue de leur intégration. PhD thesis (2005)

[Grau et al., 2008] Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P.F., Sattler, U.: Owl 2: The next step for owl. J. Web Sem. 6(4), 309–322 (2008)

[Hudelot et al., 2008] Hudelot, C., Atif, J., Bloch, I.: Fuzzy spatial relation ontology for image interpretation. Fuzzy Sets and Systems 159(15), 1929–1951 (2008)

[IGN, 2002] IGN. BDTopo Pays, version 1.2, spécification de contenu (2002)

[IGN, 2005] IGN. BDCarto, version 3, spécification de contenu, edition 1, 175 p. (2005)

[ISO, 2003] ISO. Iso 19115: Geographic information - metadata (2003)

[ISO, 2005a] ISO. Iso 19109: Geographic information - rules for application schema (2005a)

[ISO, 2005b] ISO. Iso 19110: Geographic information - methodology for feature cataloguing (2005b)

[ISO, 2007a] ISO. Iso 19131: Geographic information - draft international standard (2007a)

[ISO, 2007b] ISO. Iso 19139: Geographic information - metadata, implementation specification (2007b)

[ISO-TC-2011, 2001] ISO-TC-2011. Geographic information - rules for application schema (2001)

[Kavouras and Kokla, 2008] Kavouras, M., Kokla, M.: Theories of geographic concepts: ontological approaches to semantic integration. CRC (2008)

[Klien, 2008] Klien, E.: Semantic Annotation of Geographic Information. PhD thesis (2008)

[Lassoued et al., 2008] Lassoued, Y., Wright, D., Bermudez, L., Boucelma, O.: Ontology-based mediation of ogc catalogue service for the web - a virtual solution for integrating coastal web atlases. In: ICSOFT (ISDM/ABF), pp. 192–197 (2008)

[Levenshtein, 1966] Levenshtein, V.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady 10(8), 707–710 (1966)

[Lieberman et al., 2007] Lieberman, J., Singh, R., Goad, G.: W3C geospatial ontologies (2007)

[Mustière et al., 2009] Mustière, S., Abadie, N., Aussenac Gilles, N., Bessagnet, M.-N., Kamel, M., Kergosien, E., Reynaud, C., Safar, B.: GéOnto: Enrichissement d'une taxonomie de concepts topographiques. In: Spatial Analysis and GEOmatics Sageo 2009, Paris France (2009)

[Nambiar et al., 2006] Nambiar, U., Ludaescher, B., Lin, K., Baru, C.: The geon portal: accelerating knowledge discovery in the geosciences. In: Proceedings of the 8th Annual ACM International Workshop on Web Information and Data Management, WIDM 2006, pp. 83–90. ACM (2006)

[Partridge, 2002] Partridge, C.: The role of ontology in integrating semantically heterogeneous databases. Technical report, National Research Council, Institute of Systems Theory and Biomedical Engineering (LADSEB-CNR), Group of Conceptual Modeling and Knowledge Engineering, Padoue (2002)

[Paul and Ghosh, 2006] Paul, M., Ghosh, S.K.: An approach for service oriented discovery and retrieval of spatial data. In: Proceedings of the 2006 International Workshop on Service-oriented Software Engineering, SOSE 2006, pp. 88–94. ACM (2006)

[Reitz et al., 2009] Reitz, T., de Vries, M., Fitzner, D.: Conceptual schema specification and mapping. Technical report (2009)

[Sheth and Larson, 1990] Sheth, A.P., Larson, J.A.: Federated database systems for managing distributed, heterogeneous, and autonomous databases. ACM Computing Surveys 22, 183–236 (1990)

[Simperl, 2009] Simperl, E.: Reusing ontologies on the semantic web: A feasibility study. Data Knowl. Eng. 68, 905–925 (2009)

[Uitermark, 2001] Uitermark, H.: Ontology-Based Geographic Data Set Integration. PhD thesis, Enschede (2001)

[W3C, 2004] W3C. Owl web ontology language, overview, W3C recommendation (2004)

[Wiederhold, 1992] Wiederhold, G.: Mediators in the architecture of future information systems. IEEE Computer 25(3), 38–49 (1992)

[Yujian and Bo, 2007] Yujian, L., Bo, L.: A normalized levenshtein distance metric. IEEE Trans. Pattern Anal. Mach. Intell. 29, 1091–1095 (2007)

Methods and Tools for Automatic Construction of Ontologies from Textual Resources: A Framework for Comparison and Its Application

Toader Gherasim, Mounira Harzallah, Giuseppe Berio, and Pascale Kuntz

Abstract. Over the recent years, several approaches and tools for the automatic construction of ontologies from textual resources have been proposed. This paper provides a comparative analysis of four well known approaches and related tools among existing ones. The selected approaches and related tools indeed cover all the steps of the ontology construction process. In the first part of the paper, we introduce Methontology and related task i.e. a well-known reference methodology designed for the manual construction of ontology; then, according to Methontology, we analyze and classify detailed subtasks required by those approaches. Based on this uniform classification, we provide a very detailed comparison of those approaches: we explain the main techniques and introduce tools used in the various subtasks of each approach and we highlight the main similarities and differences between the techniques used in comparable subtasks belonging to distinct approaches. In the second part of the paper, we introduce various measures for evaluating tools effectiveness wrt a manually constructed ontology. Then, we evaluate and compare the key tools supporting those approaches by using the provided measures and a specific set of textual resources.

1 Introduction

Since the foundational work of Gruber [Gruber, 1993], ontologies are an essential element for knowledge engineering, and the development of the semantic web increased even more their importance. Today, in several domains, ontologies

Toader Gherasim · Mounira Harzallah · Pascale Kuntz
LINA, UMR 6241 CNRS
e-mail: {toader.gherasim,mounira.harzallah}@univ-nantes.fr,
pascale.kuntz@polytech.univ-nantes.fr

Giuseppe Berio
LABSTICC, UMR 6285 CNRS
e-mail: giuseppe.berio@univ-ubs.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 177–201.
DOI: 10.1007/978-3-642-35855-5_9

are considered the central component of decision support or information retrieval systems (e.g. [Osborne et al., 2009] focuses on ontologies in the medical domain and [Bourigault and Lame, 2002] focuses on ontologies in the legal domain). Earlier, ontology construction was largely based on human expertise. Today, ontologies are more and more used, tend to contain thousands of concepts (as reported in [Aime et al., 2009]) and therefore it is really interesting to massively use automation to better targeting the contribution of human experts. Indeed, as well known, for building an ontology, experts start from reference documents and knowledge by identifying the central concepts and (various kinds of) relations, often extracted from names, verbs and definitions (as also typical in the general areas of software and information system design, especially focusing on requirements). Therefore, it is very interesting to try to extract, prune and filter the huge amount of elements that can be found in input documents and general knowledge.

Accordingly, during the last decade several approaches for automatic ontology construction from text corpus (i.e. reference documents but also general documents) have been proposed [Velardi et al., 2007, Maynard et al., 2009b]. A wide range of techniques has been used to automatically execute the different tasks of ontology construction process. Because most of these approaches have been developed and tested in specific application contexts, sometimes, the employed techniques can be applied only within these contexts and provide interesting outcomes only for specific types of text content.

The aim of this paper is to propose a comparative analysis of some of these approaches and of the associated tools, implementing mentioned techniques. To provide a coherent framework for comparing approaches, we have used Methontology [Fernandez et al., 1997], one of the most known methodologies for ontology construction, supplying a set of reference tasks necessaries to build an ontology. Our interest has been concentrated towards approaches which cover all the steps of the ontology construction process as defined according to Methontology i.e. approaches that take as input texts and that propose as output an ontology in its most basic form – concepts, instances, relations. Tools associated to compared approaches are in turn compared, whenever possible and meaningful, on two distinct plans, technical and experimental, within a coherent test-bed platform.

Through the paper we consider and analyze four approaches and related tools[1]: OntoLearn – *TermExtractor*, *WCL System* [Navigli et al., 2003, Velardi et al., 2007], Alvis [Nedellec, 2006], Text2Onto – *Text2Onto* [Cimiano and Volker, 2005], and SPRAT – *SPRAT* [Maynard et al., 2009a, Maynard et al., 2009b]. These approaches construct domain ontologies that reflects the domain covered by the input texts, and not top level, highly abstract ontologies or lexicalized ontologies (like WordNet).

The paper is organized as follows. In the first part (Section 2) we present the tasks of the conceptualization activity of Methontology: this allows both a deep understanding of each approach and enables a further comparison between the four approaches mentioned above. In the same section, the various techniques employed by tools are also shortly presented. In the second part (Section 3.1), we use again

[1] The name of tools is in italics.

tasks belonging to Methontology and identified subtasks specific to each approach to precisely describe supporting tools and related supported tasks, subtasks and techniques. Then (Section 3.2) we provide an analysis of supporting tools using some of the general and extraction criteria of the framework proposed in [Park et al., 2011]. In Section 3.3 we describe our experimental tests on selected tools mentioned above (on one medium-size corpus), analyze and compare the obtained results. Section 3.4 is devoted to a critical discussion and the last section provides the reader with concluding remarks.

2 Comparison of Four Relevant Approaches for Automatic Construction of Ontologies from Textual Resources

2.1 Methotology

Methontology [Fernandez et al., 1997, Corcho et al., 2005] is one of the most known methodologies for ontology construction. It is a general, domain independent methodology, which defines the main activities of the ontology construction process and specifies the steps for performing them. The most important activity is the conceptualization, where informal knowledge is converted into semi-formal specifications which enable the identification of the main ontology components. We have reviewed the literature on Methontology and below we present the seven tasks belonging to the conceptualization activity.

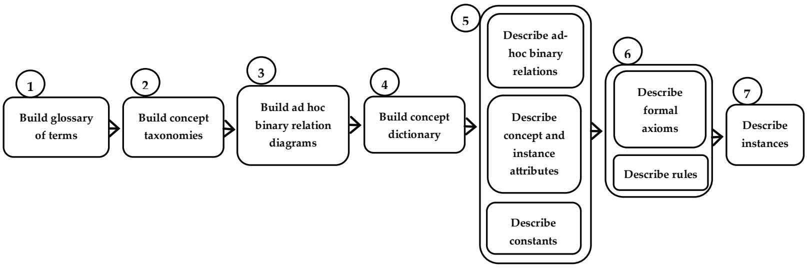


Fig. 1 Tasks of the conceptualization activity of Methontology (figure adapted from [Corcho et al., 2005])

The glossary of terms built in the **first** task (**build glossary of terms**) contains all the terms and associated definitions corresponding to the different ontological constituents (such as concept, instance, relation and attribute). In the **second** task (**build concept taxonomies**) taxonomies are constructed by selecting from the glossary the terms that are concepts, by grouping similar terms corresponding to a same concept, and by arranging concepts in a concept (*is-a*) hierarchy. The **third** task (**build diagrams for ad-hoc binary relations**) concerns binary non-taxonomic relations involving previously identified concepts, usually identified by analyzing the verbs presents in the glossary of terms. A **concept dictionary is built** in the **fourth** task. This dictionary also specifies the properties, instances and relations that are linked to each concept of the taxonomy. In the **fifth** task are identified **detailed definitions for relations, attributes and constants**. Additional **formal axioms and rules** can be defined in the **sixth** task. Finally, in the **seventh** task, a **detailed description of instances** must be provided. Tasks 5, 6 and 7 can be considered as tasks for ontology refinement, because all the central structural constituents of the ontology are already identified in precedent tasks.

In Section 2.3, Methontology and its tasks will be used as a framework for comparing, on a common base, the four approaches we mentioned in the Introduction: OntoLearn, Alvis, Text2Onto, SPRAT.

2.2 Overview of Techniques for Automating Ontology Construction from Textual Resources

Generally speaking, any approach for automating, fully or partially, ontology construction starting from textual documents comprises several algorithmic techniques that are based on theoretic and empiric principles.

[Buitelaar et al., 2005] and [Nazarenko and Hamon, 2002] have already proposed classifications of those techniques. Therefore the aim of this section is to focus and shortly present the techniques implemented in the various tools associated to the four approaches mentioned in the Introduction. These relevant techniques are useful for automating, possibly partially, the first three tasks of the Methontology conceptualization activity i.e.: (1) Build glossary of terms (and more specifically focusing on term extraction), (2) Build concept taxonomies, and (3) Identify ad-hoc relations. Other techniques mentioned in [Buitelaar et al., 2005] cover also other tasks (e.g. Describe rules) but they are however not implemented in the context of the four approaches mentioned in the Introduction. Therefore these additional existing techniques are not reported in the remainder.

Within the context of **Build glossary of terms**, term extraction is usually supported by two types of technique used jointly i.e. *linguistic techniques (L)* and *statistical techniques (S)*. *Linguistic techniques* analyze sentences and discourses in terms of grammatical constituents: delimiting terms, morphosyntactic tagging of terms (e.g. by using Noun, Verb, and Adjective), identifying syntactic constituents (e.g. subject, direct object) and relations between them (usually focusing on verbs). *Statistical techniques* are based on the frequency of a term in one document or in all the documents of one corpus. According to [Zouaq and Nkambou, 2010], these techniques include the popular term measure is *TFIDF* (i.e. normalized term frequency, inverse document frequency [Salton et al., 1975]). Definitions of terms, as the base for building a glossary, are usually found by using techniques looking to *external resources (ES)* such as controlled web pages or WordNet.

For **Build concept taxonomies** task two types of techniques are often used: *structural techniques (St)* and *contextual techniques*. Concrete subtypes of this last type of techniques are presented in the remainder. However, there are some approaches that use *external resources* (WordNet, dictionaries, etc.) rather than the text for building the taxonomy.

Structural techniques use the structure of a term representing a concept. They can be based on the syntax of the term (e.g. *domain ontology* subsumes *ontology*), on the morphology of the term (e.g. *blood mononuclear cell* as a variant of *blood cell*), on the lexical structure of the term (when the internal structure of the term serves as support for term clustering [Nazarenko and Hamon, 2002]) or on the meaning of the term (when the meaning of a complex term is built by taking into account the structure of the term and the meaning of the words that compose the term; the meaning of the words is found in external resources such as WordNet or dictionaries).

Contextual techniques are based on the context where appear terms representing concepts. A context is usually defined as a vector representing syntactic dependencies between the term that represents the concept and other surrounding terms (surrounding terms provide the context). Two main families are recognized as part of those techniques: *distributional & clustering techniques (Di & Cl)* and *pattern based techniques (Pa)*.

Distributional & clustering techniques try to cluster together concepts according to the distributions of their associated terms. These techniques are based on the hypothesis that a term has the same meaning when occurring in similar contexts ([Harris, 1968]).

Pattern based techniques try to identify in texts expressions that contain terms representing concepts and whose structure follows the given pattern. The most popular patterns for the subsumption relation are the so-called Hearst patterns ([Hearst, 1992]); for instance, the pattern *NP such as NP, NP, NP ... and NP*[2] applied to the sentence *fruits such as orange and apple* extracts that *apple* and *orange* are subsumed by *fruits*.

Patterns can be predefined or learnt. For the latter case, specifically developed *pattern learning techniques (PL)* have been introduced. These techniques are often based on a set of training concept pairs that satisfy a given relationship.

Finally, for **Identify ad-hoc relations** task, *pattern based techniques* are often used. There exists a large variety of patterns corresponding to different ad-hoc relations: *structural patterns* (e.g. *NP part-of NP*), *domain specific patterns* (e.g. *NP caused by NP*, typically referenced in the medical domain), etc. However, *pattern learning techniques*, *external resources* and *distributional & clustering techniques* are sometimes used.

Our tool analysis reveals that relevant tools often combine several techniques (i.e. resulting in kinds of hybrid techniques) for achieving better results for the purpose of the supported tasks.

[2] NP – Noun Phrase.

2.3 *Using Methontology as a Conceptual Comparison Framework*

In this section we use Methontology as a reference to compare on a common basis the tasks and the subtasks of the four selected approaches. Also, we identify and compare the main techniques employed within each approach to automatically execute the different tasks.

More precisely, we consider that the seven tasks presented in the Section 2.1 define a complete repository of tasks which must be executed in order to construct an ontology. The four approaches are mainly focused on aspects related to the construction of the structure of the ontology (identification of concepts, concepts taxonomies, relations and instances), aspects which correspond to the first four tasks of our repository, and pay little attention to the refinement of the ontology.

Table 1 summarizes the most important correspondences between the Methontology tasks and the four approaches mentioned in the Introduction: it should be noted that for each approach the task partition is based on the Methontology task definition. When possible, we also indicate in Table 1, for each task of each approach, the type of techniques that are used to automate the task. *Man* acronym is used for uniformity, to indicate a manually performed task.

3 Comparative Analysis of Tools Related to the Four Relevant Approaches for Automating Ontology Construction Process from Textual Resources

In the remainder, we focus on the tools that support the four approaches identified in Section 2. Section 3.1 describes the various tools, indicating the types of techniques they implement, and provides a general description of how tools are used within their respective approaches. Each of the next two sections is devoted to a detailed comparison of tools. The first (Section 3.2) is based on the technical features that characterize those tools. The second (Section 3.3) concerns only available tools (because, as pointed in Section 3.2, some tools are not available) and is based on a series of experimental tests.

The technical features of tools refer to characteristics and functionalities such as accepted formats, generated formats, possibilities of configuration, etc. The technical features that we have identified in a previous work ([Gherasim et al., 2011]) can be re-organized according to what in [Park et al., 2011] are named *general* and *extraction criteria*. We are going to use those criteria to present our current work (Section 3.2).

Tests are devoted to compare tools based on their outcomes. For the same purpose, [Park et al., 2011] have introduced various quality criteria, namely *semantic criteria* (comprising *interpretability*, *consistency* and *clarity* sub-criteria) and *pragmatic criteria* (comprising *accuracy*, *completeness* and *coverage* sub-criteria). Those criteria are used to evaluate, for each tool, the automatically built ontology.

Table 1 Correspondences and differences between the tasks of the four approaches and the repository of tasks inspired from Methontology

Methontology tasks	OntoLearn tasks		Alvis tasks	Text2Onto tasks	SPRAT tasks
Build a glossary of terms *Identify and define terms corresponding to the different elements of the ontology: concepts, attributs, instances and relations*	Terminology extraction – *L & S* Terminology filtering – *using L & S filters* Terminology validation – *Man* Identification of definitions for terms *(which will compose a glossary) – using Internet searches – ES* *Terms = { concepts }*		Terminology extraction – *L* Terminology validation – *Man* *Terms = { concepts, instances }*	Terminology extraction – *L & S* *Terms = { concepts, instances }*	Terminology extraction – *L* *Terms = { concepts, instances }* **Iterative execution** Choose a term that is involved in a *'is-a'* pattern that links him to a concept that already exists in the ontology – *Pa*
Build concept taxonomies *Group terms that correspond to the same concept* *Arrange in one or more taxonomies the already identified concepts*	Semantic disambiguation of every complex term by intersecting semantic nets associated to each word composing the term: *SSI algorithm (specific to OntoLearn) and WordNet – ES & St* Finding taxonomy relations on the synsets associated to each word composing a complex term underlying concept	Hypernym Extraction Star pattern identification and sentence clustering – *Man* Word-Class Lattice (WCL) construction – *PL* Sentences matching with WCL – *Pa*	Build a taxonomy – *Identify the contextual attributs of each term; build a taxonomy using the unsupervised classification (based on terms attributes) of terms and classes of terms (concepts) – Di & Cl*	Identify *'is-a'* relations between concepts *(using WordNet, patterns and heuristics (based on the structure of compound terms) – St & Pa & ES*	*(if the chosen term is a concept)* Add the concept to the ontology *– using rules for inserting concepts in the ontology and rules for resolving conflicts (these rules are specific to SPRAT) – Pa*
Build diagrams for ad hoc binary relations *Group terms that correspond to the same relation* *Define binary relations by identifying, for each relation, the related concepts – by analyzing the pair of terms associated with each verb (its subject and its object)*	Provide domain semantic relations examples (learning set) – *Man* Learn rules to classify the relations that hold between pairs of concepts – *PL* Apply the rules to complex term to identify relations between its components – *Pa*		Identify the syntactic dependencies that can properly characterize a relation – *using ASA (Abstraction of the Syntactic Analysis), a formalism specific to Alvis* Identify some examples – *Man* Learn a set of rules for discovering relations – *using an inductive learning program that uses the ASA formalism – PL* Identify concepts connected by the relations learned from examples – *automatically, using rules – Pa*	Identify *'subtopic-of'* relations between concepts *– (using statistical analysis of cooccurrence of concepts) – Di & Cl* Identify general relations (relations defined by verbs) *– (using a shallow parsing strategy and information about the frequency of the terms) – Di & Cl*	Identify and add to the ontology relations between the new concept and concepts that already exist in the ontology *(using predefined patterns) – Pa*
Build the concept dictionary *(that contains, for each concept, the associated attributes, instances and relations)*	—		—	Identify *'instance-of'* relations – *Di & Cl*	*(if the chosen term is an instance)* Add the instance to the ontology *– using specific SPRAT rules – Pa*

Human-experts are asked to assess, based on the provided criteria definitions, the quality of ontologies according to each criteria. [Park et al., 2011] tried to limit the subjectivity by asking four human-experts to evaluate each ontology. However, after a deep analysis, we consider that some of the *semantic* and *pragmatic criteria* as well as the associated evaluation methods proposed in [Park et al., 2011] are not fully suitable for a precise comparison between tools.

On the one hand, according to our previous work ([Gherasim et al., 2011]), we consider that a more precise comparison among tools outcomes should be performed by using a common reference ontology: the various distinct ontologies built by using tools can then be compared to that single common reference ontology. On the other hand, despite the interest of *semantic* and *pragmatic criteria* introduced in [Park et al., 2011], we consider that some of those criteria/sub-criteria are vague or not suitable for evaluating ontologies automatically built by tools. Specifically, we consider that *consistency* remains vague (indeed the term "consistency" in the context of ontologies may be interpreted as "logical consistency" it should also be noted that the term "consistency" in the context of ontologies may be interpreted as "logical consistency"), *clarity* does not take into account the ontology domain and *interpretability* focuses only on WordNet; *completeness* and *accuracy* are suitable sub- criteria but their associated evaluation methods, only based on expert judgment, are not suitable especially because ontologies are automatically built (so ontologies lack of concept definitions and the number of concepts may growth exponentially with text-size; also, relationships, when available, are very intricate).

Therefore, in our current work, any tool comparison based on tool outcomes is performed as in our previous work ([Gherasim et al., 2011]). Specifically, a common reference ontology has been manually built (a general golden standard is rarely

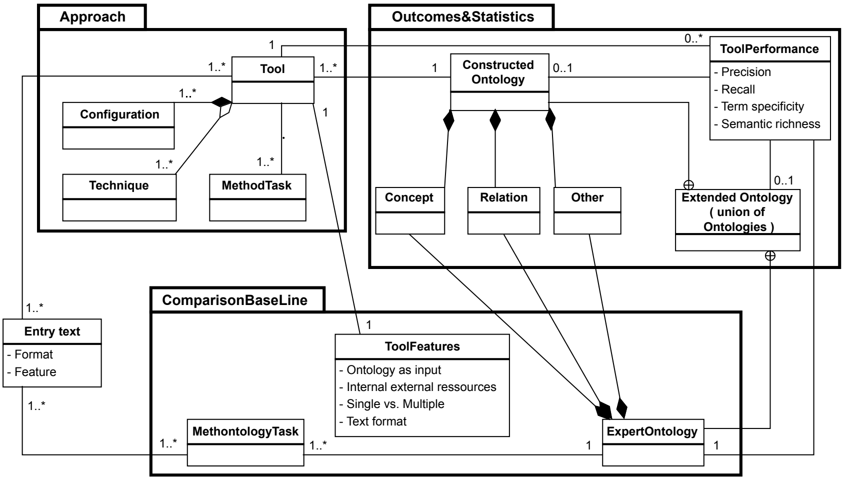


Fig. 2 The UML diagram of the test-bed platform used for performing the different tests

available, see Section 3.3). Standard measures (precision and recall) and additional measures (term specificity and semantic richness) are introduced and used to compare ontologies automatically built by tools to the common reference ontology. The manually built ontology reflects the expert knowledge about the specific domain, as restrained by the texts (a partially view on the domain) submitted to tools. Section 3.3 presents in detail the results of the various tests.

A test-bed platform has been realized for performing tests according to the discussion above. Figure 2 shows the platform packages as a UML diagram. The *Approach* package is used to describe the approach in detail, comprising *Tool*, *Task*, *Technique* and *Configuration* (*Configuration* refers to how a tool is configured and installed within the platform). The *ComparisonBaseLine* package provides the reference ontology (manually built according to Methontology); the package also comprises the technical features of tools. The *Outcomes&Statistics* package provides the various ontologies built by tools and the standard measures used to compare those ontologies to the reference ontology; the package also comprises the extended ontology which is built as the union between the reference ontology and ontologies built by tools. The reason for including this union is discussed in Section 3.3.1. Finally, *EntryText* is used to describe the input corpus. It is characterized by its format (such as plain text, HTML text, etc.) and by some features (such as *dense*, *self-contained*, etc.). Although the text features are extremely important, in this paper we have not fully exploited them because such an analysis requires a large number of additional tests.

3.1 Analysis of the Role of Each Tool for the Corresponding Approach

Each approach is supported by one tool or a set of tools that implements the various techniques presented in Section 2.2. Table 2, built on Table 1, lists those tools and highlights the correspondences between them and the different tasks/subtasks belonging to the identified approaches.

Table 2 is therefore used to partially instantiate the package *Approach* in our test-bed platform (see Section 3 introduction and Figure 2). There are subtasks for which no tool is proposed (e.g. “Provide domain semantic relations examples” subtask of OntoLearn approach), or it remains unclear if some tools are available (e.g. “Identify relations” subtask of Alvis approach).

There are subtasks for which several tools are proposed (e.g. “Terminology extraction” subtask of SPRAT approach). Some tools correspond to only one specific subtask, such as *GlossExtractor*; some tools cover two or several subtasks, or even the entire ontology construction process (e.g. *SPRAT*).

Table 2 also allows to identify comparable tools: for instance, it is not possible to compare *TermExtractor* with the *Relation* module of *Text2Onto*, but it is possible to compare *TermExtractor* with the *Concept* module of *Text2Onto*.

Table 2 Correspondences, for each approach, between its subtasks and its supporting tools. Tested tools are highlighted. For each tool we indicate in parentheses which type of techniqes it implements.

Methontology tasks	OntoLearn tasks				Alvis tasks		Text2Onto tasks		SPRAT tasks			
	Subtask		Tool		Subtask	Tool	Subtask	Tool	Subtask	Tool		
Build a glossary of terms	Terminology extraction Terminology filtering		**TermExtractor** *– (L & S)*		Terminology extraction	*YATEA – (L)*	Concept extraction	**Text2Onto** *module concept – (L & S)*	Terminology extraction	*JAPE, NEBOnE – (L)*	*Term Raider – (L & S)*	**SPRAT** *– (Pa)*
	Terminology validation		—		Terminology validation	—	Instance extraction	**Text2Onto** *module Instance – (L & S)*	Choose a term		—	
	Identification of definitions for terms		*GlossExtractor – (ES)*									
Build concept taxonomies	Semantic disambiguation of complex terms	Hypernym Extraction	*SSI – (ES & St)*	**WCL System–** *(PL, Pa)*	Build a taxonomy	*BioLG, ASIUM adapted for Alvis – (Di & Cl)*	Identify *’is-a’* relations	**Text2Onto** *module SubclassOf – (St & Pa & ES)*	Add the concept to the ontology		—	
	Find taxonomy relations		—									
Build diagrams for ad hoc binary relations	Provide domain semantic relations examples		—		Abstraction of the Syntactic Analysis	*LINK-PARSER*	Identify *’subtopic-of’* relations	**Text2Onto** *module SubtopicOf – (Di & Cl)*	Identify and add relations to the ontology		—	
	Learn rules		*C4.5 – (PL)*		Identify some examples	—						
	Identify relations between the components of complex terms		—		Learn a set of rules	*Propal – (PL)*	Identify general relations	**Text2Onto** *module Relation – (Di & Cl)*				
					Identify relations	—						
Build the concept dictionary	—		—		—	—	Identify *’instance-of’* relations	**Text2Onto** *module InstanceOf – (Di & Cl)*	Add the instance to the ontology		—	

We can classify these tools in two groups: in the first group there are generic tools that were developed for another purpose than building ontologies, and have been found useful for building ontologies; in the second one, there are specialized tools especially designed for ontology construction.

The first group includes programs that can manipulate and identify regular expressions in texts (e.g. *JAPE* [Maynard et al., 2009a, Maynard et al., 2009b]), syntactical analyzers (e.g. *LinkParser*, *BioLG* [Nedellec, 2006]), inductive learning programs (e.g. *C4.5* [Navigli et al., 2003], *Propal* [Nedellec, 2006]) and term extractors (e.g. *YATEA* [Nedellec, 2006]).

The second group contains specialized tools that were developed to support the ontology construction process and which are directly associated with the previously presented approaches: *Text2Onto* [Cimiano and Volker, 2005] for the approach with the same name; *TermExtractor*, *GlossExtractor* [Velardi et al., 2008], *SSI* [Navigli and Velardi, 2005] and *WCL System* for OntoLearn; *NEBOnE*, *TermRaider* and *SPRAT* [Maynard et al., 2009a] for SPRAT; *ASIUM* [Nedellec, 2006] for Alvis.

We note that Text2Onto proposes a specialized tool (*Text2Onto*) that covers the entire process of extracting an ontology. SPRAT proposes several tools, but one of them, that has the same name as the approach – *SPRAT*, covers the entire process of extracting an ontology. *SPRAT* as tool uses, in a transparent manner, the results of *JAPE* and *NEBOnE*, but it does not reuse *TermRaider* results. OntoLearn and Alvis do not propose any tool covering the entire process but a set of tools where each tool covers partially the process; a tool can take as input the results of another tool and, eventually, the set of tools covers the entire process of extracting an ontology.

OntoLearn provides specialized tools for building a term glossary (*TermExtractor*, *GlossExtractor*) and the concept taxonomy (*WCL System*), but for the relations extraction most of the works remains to be manually performed. Nevertheless, users can be assisted by a tool (*C4.5*) that learns rules for discovering relationships over compound terms underlying concepts (such as a rule establishing that in a compound term *XY*, if *X* is a type of building material, then *X MATTER Y* holds, with confidence 0.5, being *MATTER* a relation – applying this rule to *stave church*, *Church MATTER Stave* is a possible relation). Alvis proposes a generic tool (*YATEA*) for building a term glossary, a specialized tool (*ASIUM*) for the taxonomy construction, and similarly to OntoLearn, a tool (*Propal*), enables rules learning from examples for ad-hoc relation extraction subtask.

3.2 Tool Comparison Based on Technical Features

This section provides a tool comparison based on technical features, defined as *general* and *extraction criteria* in [Park et al., 2011]. The content of this section is used to partially instantiate the package *ComparisonBaseLine* within the test-bed platform.

The **general features** deal with the exterior features of tools: user interface, availability and time to first use. As in our previous work, we are interested only in the tools' **availability**, defined as *if a tool can be acquired without too much effort*. We adapted this criterion to take into account one of the specificity of the tools we analyze – the fact that some of them are available as web services. So, our definition for tool's availability is: *if a tool can be acquired and installed on local machines or if it is available as a web service or web application and can easily be accessed and tested.*

The tools proposed by OntoLearn, *TermExtractor*, *GlossExtractor* and *SSI*, are available as web applications, on the authors' servers. They can be accessed via a webpage. The development of *WCL System* has just finished, so that *WCL System* is not yet released and it can be tested only by asking the authors to test it. For Alvis, *ASIUM*, the only specialized tool proposed, is not available to be tested. *Text2Onto* can be downloaded and tested on a local computer. *TermRaider* and *SPRAT*, the tools proposed by SPRAT, are available only as web services. The availability of all these tools, except *Text2Onto*, is closely related to the availability of the web servers where they are hosted and we met some access difficulties in our experimentations.

Because *ASIUM* is not available to be tested and *YATEA* is just a generic tool for term extraction we think that it is uninteresting to keep in our analysis Alvis and its tools. So, from now, we will ignore them. For the same reason we ignore *C4.5*, the generic tool used by OntoLearn to learn rules for relation extraction.

The **extraction features** concern the main function used for ontology extraction: (1) preprocessing requirement; (2) ontology reuse; (3) extraction level; (4) degree of automation; (5) algorithm selection; (6) efficiency; (7) reliability. Another extraction feature, not included in [Park et al., 2011] framework, concerns the auxiliary tools and external resources (8) that are used by each tool.

The first criterion, (**preprocessing requirement** (1)), consider whether a tool requires or not additional preprocessing of documents taken as input (e.g. linguistic annotation). This criterion partially corresponds to a criterion from our previous work – the *type of inputs and outputs*. As we here also analyze tools that do not take texts as input, we adapt the *preprocessing requirement* criterion in order to take into account also the type of inputs of each tool, and not only the preprocessing effort.

All the tools covering initial subtasks of each approach (*TermExtractor*, *Text2Onto*, *TermRaider* and *SPRAT*) accept as input simple text files (txt). *Text2Onto* also accepts PDF files, and *TermExtractor* PDF, DOC, HTML files or archives containing this type of files. *SSI* and *GlossExtractor* take as input a list of terms, and *WCL System* a list of definitions. No one of all these tools needs preprocessing efforts.

The second criterion (**ontology reuse** (2)) takes into account the fact that a tool can use concepts from existing ontologies or an entire ontology when constructing a new one. As for the precedent criterion, we extend this criterion to take into account the fact that there are tools that take as input list of terms or definitions. *TermExtractor* and *TermRaider* can enrich an existing list of terms with new terms. *Text2Onto* can selectively update its results when the texts input evolve. *SPRAT* can take as input an ontology and enrich it with new concepts.

The **extraction level** (3) examines whether a tool can automatically or semi-automatically extract concepts or both concepts and their relations. We further refine this criterion by further differentiating between taxonomic relations and the other relations and by taking into account the presence/absence of instance identification.

For the three approaches, the corresponding tools (*TermExtractor*, *Text2Onto* and *SPRAT*) are able to identify concepts. Only *Text2Onto* and *SPRAT* can identify instances. *WCL System*, *Text2Onto* and *SPRAT* can identify taxonomic relations. Only *Text2Onto* and *SPRAT* can automatically identify other (ad-hoc) semantic relations.

The fourth criterion concerns the **degree of automation** (4) for extracting concepts and relations: a tool can be considered automatic if it performs that extraction without any human intervention; a tool can be considered as semi-automatic if the user must supply some extraction rules or whenever any human intervention is required during that extraction.

Text2Onto and *SPRAT* are automatic tools. The four tools proposed by OntoLearn (*TermExtractor*, *GlossExtractor*, *SSI* and *WCL System*) are automatic when considered independently, but, they are not integrated in a fully automatic system for ontology construction.

The fifth criterion, called **algorithm selection** (5), takes into account the possibility for users to select various algorithms/techniques when using one tool. This criterion partially corresponds to the *configurability* criterion of our previous work. We extend it by accounting the possibility to configure different parameters of the proposed algorithms.

Only two tools, *TermExtractor* and *Text2Onto* are configurable. *Text2Onto* provides several algorithms, targeting extraction of concepts, instances, relations; users can select an algorithm or apply a strategy to combine different results. *TermExtractor* proposes several parameters like the maximum number of terms in a compound term, different filtering thresholds and allows to measure and to use the position and the emphasis of the words in the textual analysis.

The sixth criterion is the **efficiency** (6), which measures the convergence speed. *Text2Onto*, *SPRAT* and *TermRaider* seem to be very efficient tools: in our tests (Section 3.3), results have been obtained in less than one minute. *TermExtractor*, *GlossExtractor* and *SSI* are only available through a web interface for submitting inputs and therefore tools are executed depending on server load. For this reason, it is very difficult to precisely evaluate the tool efficiency. *WCL System* is not directly available but you can request authors to process your tests.

The **reliability** (7) criterion reviews if the output remains consistent over repeated tests with the same input data. All the tools, excepting *GlossExtractor*, are reliable. Indeed, as *GlossExtractor* depends on web searches, consistency between results cannot be guaranteed.

Concerning the last criterion – **auxiliary tools and external resources** (8) that are used by each tool – *Text2Onto* requires *GATE*[3] and *TreeTagger*[4] as auxiliary tools and WordNet[5] as auxiliary resource. *SSI* also uses WordNet and *GlossExtractor* use Internet searches. As *SSI*, *TermExtractor* and *GlossExtractor* are already installed on Web servers and ready for use, no auxiliary tool is required. To test

[3] http://gate.ac.uk/, version 4.0.

[4] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger

[5] http://wordnet.princeton.edu/, version 2.0.

TermRaider and *SPRAT*, which are available as Web services, a specific plugin[6] for *Neon Toolkit*[7] must be installed.

3.3 Tool Experimental Comparison

As defined in the introductory part of Section 3, the experimental comparison focuses on comparing tools by conducting a set of tests within the test-bed platform. According to the test-bed platform, tools outcomes are not only influenced by the implemented techniques but also by features of input texts (or input corpus). However, we do not discuss here in detail these features because requiring additional work. This additional long term work is discussed in the concluding section.

3.3.1 Experimental Settings

The set of tests has been organized as follows. The first phase focuses on concepts and instances, while the second phase focuses on taxonomic relations. Tests for ad-hoc relations have also been performed in a third phase but not presented in this paper because outcomes have been largely meaningless. We have compared the relevant tools outcomes with a common reference ontology (O_{mb}), manually built by following the tasks of Methontology (Fig. 3).

In order to take into account the imperfections of this common reference ontology, we have introduced an adjustment process: the expert (who has manually built the ontology) can validate further ontology elements (concepts, instances and taxonomic relations) that are comprised in one of the automatically built ontologies but are not part of O_{mb}.

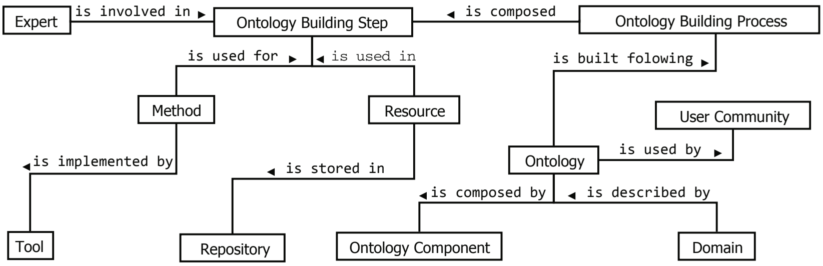


Fig. 3 The core concepts of the manually built ontology and their main relations

The validation of ontology elements by the expert is based on an adapted version of the framework ([Volker and Sure, 2006]) developed to evaluate the results of *Text2Onto*. In the adapted version, to each extracted ontology element the expert

[6] http://neon-toolkit.org/wiki/Gate_Webservice, version 1.1.15.

[7] http://neon-toolkit.org

assigned a score between 1 (fully invalid) to 4 (fully valid) according to the following scale:

- Concepts / Instances
  - 1 - terms that are not concepts or instances (e.g. *non*, *cannot*, *one*, *creating*)
  - 2 - terms that are more like instances than concepts / concepts than instances (e.g. *OntoLearn*, *EuroWordNet project* / *concept*, *ontology*)
  - 3 - terms that identify concepts / instances irrelevant for our domain (e.g. *european project*, *research institution* / *Alfonseca*, *Vossen*)
  - 4 - terms that identify concepts / instances relevant for our domain (e.g. *concept*, *ontology* / *OntoLearn*, *WordNet*)
- Taxonomic relations
  - 1 - fully incorrect: the relation is not correct, or one of the terms it relies is not a concept or an instance (e.g. *term* is a *figure*)
  - 2 - correct to some extent: the two terms/concepts are related but the relation is a true taxonomic relation only in restricted contexts (e.g. *task* is a *project*)
  - 3 - correct: a true taxonomic relation where at least a concept is, referring to the domain, too general or specific (e.g. *tool* is a *object*)
  - 4 - fully valid: a correct relation which relies two domain relevant concepts (e.g. *linguistic processor* is a *tool*)

The validation process has allowed us to account that one tool can provide additional ontology elements that the expert did not include in he first version of O_{mb} but to which he assigned a score of 4. The union ontology (O_u), part of the test-bed platform, is constructed by extending O_{mb} with all these additional valid ontology elements.

As indicated in the test-bed, this adjustment process led us to compare the automatic outputs not only with O_{mb} but also with O_u. Accordingly, we calculated two values for precision and recall: $Precision_m$, $Precision_u$ and $Recall_m$, $Recall_u$. The first value corresponds to the comparison with O_{mb} and the second value to the comparison with O_u. These two measures are computed as following for each type of ontology elements:

$$Precision_m = \frac{|EE \cap O_{mb}|}{|EE|} \qquad Recall_m = \frac{|EE \cap O_{mb}|}{|O_{mb}|}$$

$$Precision_u = \frac{|EE \cap O_u|}{|EE|} \qquad Recall_u = \frac{|EE \cap O_u|}{|O_u|}$$

where $EE = the\ set\ of\ Extracted\ Elements$ $\qquad |X| = the\ number\ of\ elements\ of\ X$

Two other measures of the test-bed platform allows us to compare the results of the different tools with O_{mb}: (1) the *term specificity* and (2) the *semantic richness* of a taxonomic relation.

The first measure, *term specificity*, is based on the idea that complex terms composed of two, three or even more words are more likely to correspond to specific domain concepts than simple terms (one word terms). In our analysis, we consider three levels of specificity, corresponding to the number of words in one term: one word, two words and three or more words.

The second measure, the *semantic richness*, is based on the idea that a taxonomic relation linking a compound term with a simpler term lexically included in the former, is less semantically rich than a taxonomic relation linking two terms that are not lexically included one in the other one. Two values are possible for this measure: semantically rich (e.g. *ontology* is a *conceptualization*, *concept tree* is a *hierarchy*) and semantically poor (e.g. *concept tree* is a *tree*).

The set of tests is performed on a medium-size corpus (about 4000 words) that covers the domain of *ontology construction from texts*. This corpus is composed of just one document – a shrunken version of a scientific paper by Navigli and Velardi titled: *"Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites"* [Navigli and Velardi, 2004]. In fact, as most of the tools take as input plain text files, we have eliminated from the original document all the images, diagrams, tables, mathematical formulas, examples, references, etc. We have finally obtained a corpus that is very *dense* (i.e. contains a high number of concepts when compared to the number of words in the text) (see Table 3), instantiating the *Class EntryText* within the test-bed platform.

In the set of tests, we have tested *TermExtractor*, *Text2Onto* and *SPRAT*. We have excluded *SSI* (the tool proposed by OntoLearn) and *TermRaider* (the tool proposed by SPRAT) from our tests. Indeed, the result of *SSI* is a semantic net (providing the meaning of the identified concepts) that is not exploitable by hands. The required input of *TermRaider* is a corpus containing at least two distinct files and the results are very dependent on the content of the two files so that cutting one file in two (or several) distinct file generates each time very distinct results. In addition, the results of *TermRaider* are not systematically used by the other modules embedded in *SPRAT* ([Gherasim et al., 2011]).

We have decided to keep any default configuration of tools, whenever tools enable various possible configurations (i.e. *Text2Onto* and *TermExtractor*).

This choice instantiates within in the test-bed platform, the *Class Configuration* – part of the *Approach* package. When *Text2Onto* is tested using its default configuration all the proposed extraction algorithms are executed and the final result is the union of results of each algorithm. In the case of TermExtractor we kept the default values for all its parameters.

Moreover, the set of tests showed that *SPRAT* was not well-adapted to the experimental protocol. Consequently, in our current work, as reported in the paper, we propose additional tests for *SPRAT* (Section 3.3.4).

We have also tested *WCL System* on a subset of valid terms extracted by *TermExtractor* (197 terms, see Table 5). As *WCL System* is not publicly available (Section 3.2), we have asked the authors of OntoLearn to test it for us.

According to OntoLearn approach, *WCL System* works on term definitions, provided in some ways or identified by using *GlossExtractor*. For efficiently performing tests, the expert selected a representative subset of them (36 terms). The choice of these 36 terms was basically subjective but trying to keeping a similar distribution of lengths, as for the extracted terms by *TermExtractor* (see Table 4).

3.3.2 Analysis of Test Results: Concepts and Instances

In this section, we present the results of tests performed on the three tools that identify terms corresponding to concepts and instances. *Text2Onto* and *SPRAT* are supposed to identify, in separate lists, concepts and instances. *TermExtractor* identifies only concepts. With the given corpus, *SPRAT* results in a very limited ontology, containing only 9 concepts, and no instances (indeed as said above, larger tests have been performed on *SPRAT* as explained in Section 3.3.4). However, in order to preserve the uniformity of our comparisons, we have kept the results of *SPRAT* in the various analysis tables presented in the remainder.

Table 3 shows the results obtained through the validation of tools outcomes by the expert. We observe that 77% of the concepts extracted by *TermExtractor* have been evaluated as concepts relevant for domain and only 61% of the concepts, and respectively 21% of the instances extracted by *Text2Onto* have been evaluated as relevant for the domain.

Table 3 Automatically extracted concepts and instances: the results of the expert validation

	Concepts		
Score	*TermExtractor**	*Text2Onto*	*SPRAT*
All	253	444	9
1	22	30	3
2	4	2	3
3	30	138	0
4	197 (77%)	274 (61%)	3 (33%)

	Instances	
Score	*Text2Onto*	*SPRAT*
All	94	0
1	13	0
2	10	0
3	51	0
4	20 (21%)	0 (0%)

* *TermExtractor* extracts only concepts.

Table 4 compares, using the *term specificity* measure, the results of tools to O_{mb}. It can be easily seen that *Text2Onto* results in simpler terms than in the case of *TermExtractor* (simpler terms comprise less words).

Table 5 provides the double comparison of tools outcomes with O_{mb} and O_u. *TermExtractor* obtains a high quality precision: 59% and respectively 78% of the extracted terms correspond to valid concepts of O_{mb}, and respectively of O_u. Referring to O_u, *Text2Onto* has highest recall (47%), but when referring to O_{mb} it has almost the same recall as *TermExtractor* i.e.: 35% vs. 36% respectively. These figures on the medium-size corpus confirm the ones obtained on a smaller corpus ([Gherasim et al., 2011]). However, it is interesting to note that the differences between *TermExtractor* and *Text2Onto* (on precision, recall and term specificity) are smaller when working with a medium-size corpus.

Table 4 Automatically extracted concepts: comparison with O_{mb} based on the *term specificity* measure

Number of terms	O_{mb}	*TermExtractor*	*Text2Onto*	*SPRAT*
All	417	253	444	9
One word	81 (19%)	41 (17%)	311 (70%)	3 (33%)
Two words	220 (53%)	170 (66%)	116 (26%)	3 (33%)
Three or more words	116 (28%)	42 (17%)	17 (4%)	3 (33%)

Table 5 Automatically extracted concepts and instances: comparison with O_{mb} and O_u

	O_{mb}	*TermExtractor*	*Text2Onto*	*SPRAT*	O_u
Extracted concepts	–	253	444	9	–
Valid concepts	417	197	274	3	581
$\cap\, O_{mb}$	–	149	146	3	417
$Precision_m$	–	59%	33%	30%	–
$Precision_u$	–	78%	62%	30%	–
$Recall_m$	–	36%	35%	0.7%	–
$Recall_u$	69%	34%	47%	0.5%	–
Extracted instances	–	–	94	0	–
Valid instances	38	–	20	0	40
$\cap\, O_{mb}$	–	–	18	0	38
$Precision_m$	–	–	19%	0	–
$Precision_u$	–	–	21%	0	–
$Recall_m$	–	–	47%	0	–
$Recall_u$	95%	–	50%	0	–

3.3.3 Analysis of Test Results: Taxonomic Relations

In this section, we present the results of tests performed on *Text2Onto*, *SPRAT* and *WCL System* to identify taxonomic relations.

As explained in Section 3.2, these three tools do not work with the same type of input and external resources. *Text2Onto* takes as input the corpus and use WordNet as external resource. *SPRAT* takes as input the corpus and do not use any external resource. *WCL System* takes as input a *list of definitions* corresponding to terms identified by *TermExtractor*. In our tests, this *list of definitions* has been constructed by *GlossExtractor* which use Internet searches as external resource(s).

Table 6 provides the results of the expert validation of the taxonomic relations extracted by *Text2Onto* and *SPRAT*. *Text2Onto* identified 362 taxonomic relations where 111 have been evaluated as fully valid. These valid relations are between 150 concepts, out of 444 concepts already extracted and also out of 274 concepts already validated (see Table 5). *SPRAT* results are very limited: only 5 taxonomic relations are identified, and none of them is fully valid. Consequently, next further analysis presented in this section concerns only *Text2Onto* and *WCL System*.

Table 7 provides the double comparison, based on precision and recall, of *Text2Onto* results with O_{mb} and O_u.

Table 6 Automatically extracted taxonomic relations: the results of the expert validation

Score	*Text2Onto*	*SPRAT*
All	362	5
1	78	2
2	69	3
3	104	0
4	111 (30%)	0 (0%)

Table 7 Automatically extracted taxonomic relations: comparison with O_{mb} and O_u

	O_{mb}	*Text2Onto*	O_u
Extracted relations	–	362	–
Valid relations	407	111	473
$\cap\, O_{mb}$	–	45	407
$Precision_m$	–	41%	–
$Precision_u$	–	31%	–
$Recall_m$	–	10%	–
$Recall_u$	86%	23%	–

Because *WCL System* has been tested on a subset of valid terms (Section 3.3.1) it is not possible to directly compare it with *Text2Onto*, O_{mb} and O_u by using standard precision and recall.

As test outcomes over the 36 terms used with *WCL System*, the tool has identified 278 taxonomic relations involving the 36 original terms but also 246 additional terms, found by using the definitions provided by *GlossExtractor*: 67 out of 246 belong to O_u. Furthermore, only 25 of these 278 taxonomic relations have been evaluated as fully valid. These 25 relations involve only 38 concepts belonging to O_u.

To perform a comparison standard precision and recall for relations have been slightly adapted: common concepts linked by valid relations (in the case of *Text2Onto* and *WCL System*) or relations (in the case of O_{mb} and O_u) are used in the measures as explained below. Indeed, a comparison performed as above, provides useful insights about the ability of each tool to find-out more or less relations between a set of common concepts as well.

We have observed that there are 21 concepts in common i.e. concepts common among the 38 concepts linked by the 25 valid taxonomic relations found by *WCL System*, the 150 concepts linked by 111 valid taxonomic relations found by *Text2Onto*, the 417 concepts of the 407 taxonomic relations of O_{mb} and the 462 concepts of the 473 taxonomic relations of O_u. Table 8 presents this suggest comparison based on common concepts.

Accordingly, adapation of precision and recall measures are computed as follow:

$$Precision_m = \frac{|Select(EE:CC) \cap O_{mb}|}{|Select(EE:CC)|} \quad Recall_m = \frac{|Select(EE:CC) \cap O_{mb}|}{|Select(O_{mb}:CC)|}$$

$$Precision_u = \frac{|Select(EE:CC) \cap O_u|}{|Select(EE:CC)|} \quad Recall_u = \frac{|Select(EE:CC) \cap O_u|}{|Select(O_u:CC)|}$$

where $CC = the\ subset\ of\ 21\ common\ concepts$
and $Select(A:B) = all\ A\ involving\ only\ concepts\ found\ in\ B$

Table 8 Automatically extracted taxonomic relations: comparison on a subset of 21 common concepts

	O_{mb}	*WCL System*	*Text2Onto*	O_u
(Extracted relations)	–	11	7	–
(Valid relations)	6	9	5	13
$\cap O_{mb}$	–	3	3	6
∩ *WCL System*	3	–	3	9
∩ *Text2Onto*	3	3	–	5
$Precision_m$	–	27%	43%	–
$Precision_u$	–	82%	71%	–
$Recall_m$	–	50%	50%	–
$Recall_u$	46%	69%	38%	–
Specific relations*	16%	38%	8%	–

*Relations identified by only one tool.

Table 8 clearly shows the complementarity that may exist between the tools and O_{mb}, and also between the tools. Each tool identifies valid taxonomic relations that are not identified by the expert (in Table 8, we named these relation 'Specific relations'). *WCL System* seems to be really remarkable because 38% of all the relations identified between the 21 common concepts are specific to it.

The results of *WCL System* and *Text2Onto* may also seem complementary when they are analyzed alongside the *semantic richness* measure. In fact, 76% of the 111 fully valid relations identified by *Text2Onto* are semantically poor relations, while 76% of the 25 fully valid relations identified by *WCL System* are semantically rich relations.

3.3.4 Testing SPRAT with a Different Strategy

As said in the previous sections, our experimentations have shown quite limited results for *SPRAT*. To go deeper into the analysis, we have further tested *SPRAT* by the following experimental process. As *SPRAT* can take an ontology as input and enrich this ontology with new concepts, we have tested *SPRAT* with a seed ontology (containing a selection of concepts of the manually built ontology) and on the corpus. This test has been repeated with different seed ontologies for verifying if the content of seed ontology interacts in any manner with the *SPRAT* ontology extraction process. Each time, the built ontology has been a merging of the seed ontology with the ontology automatically built by *SPRAT* directly on the corpus – i.e. without using any content of the seed ontology. This fact makes us to conclude that the seed ontology has no influence on the *SPRAT* ontology extraction process.

Finally, we have also tested *SPRAT* by progressively increasing the corpus size – from 4000 words to 27000 words. Each new corpus has been iteratively constructed by adding a new content to the previous corpus. The results are presented in Table 9. The percentage of domain concepts increases with the size of the corpus. But, the built ontology stays quite flat – with a maximal depth of 2 for all the tests. Moreover, neither instance nor relation – except taxonomic relations – have been identified.

Table 9 The main characteristics of ontologies construted by SPRAT when the size of the texts has gradually increased

Corpus size Score	4000*	9000*	15000*	19000*	27000*	8000*
All	9	14	20	27	49	27
1	3	3	4	4	6	5
2	3	3	3	3	3	1
3	0	4	7	10	9	1
4	3 (30%)	4 (29%)	6 (30%)	10 (37%)	31 (63%)	20 (74%)

*The number of words in the corpus.

We have noted much better recall and precision for *SPRAT* when passing from 19000 words corpus to 27000 words corpus. This fact conducted us to test *SPRAT* on a corpus containing only the 8000 words text that has been added to the 19000 words corpus for obtaining the 27000 words corpus. *SPRAT* obtained outstanding results on this 8000 words corpus: it extracted the same number of concepts than when it has been tested on the 19000 words corpus but with better precision (74% compared to 37%).

This performance improvement is explained by the fact that the 8000 words corpus contains more expressions that match *SPRAT* predefined patterns (see Table 2) than the 19000 words corpus. When tested on the 8000 words corpus *SPRAT* extracted terms/concepts that are part of expressions containing keywords like *'such as'* or *'is kind of'*).

3.4 Discussion

Our new experimentations at a medium scale confirm that the different tools are able to scale for medium-size text corpus. Moreover, they show that *SPRAT* significantly improves its results when it takes as input medium text or text containing expressions that match specific patterns. However, *SPRAT* results seem to be of lower quality than those provided by *Text2Onto* and *TermExtractor + WCL System*.

More precisely, concerning concept-instance, *Text2Onto* and *TermExtractor* continue to have very good results as already observer in [Gherasim et al., 2011] for smaller text size; specifically, *Text2Onto* has a better recall and *TermExtractor* a better precision but differences are smaller for medium-size corpus.

Together, *TermExtractor* and *Text2Onto* identified 388 terms considered fully valid by an expert independent evaluation and corresponding to 66% of the concepts of a union ontology (the manually built ontology corresponds to 72% of the concepts of the union ontology) defined in Section 3.3.2. Additionally, 83% of the terms identified by *TermExtractor* are complex terms that are likely to correspond to specific domain concepts, while 70% of the terms identified by *Text2Onto* are simple terms underlying general concepts.

Concerning relations, we have restricted ourselves mainly to *WCL System* and *Text2Onto* as we have shown the results of *SPRAT* are very limited under the same test case. Both *WCL System* and *Text2Onto* identify taxonomic relations. Since *WCL System* is not yet released and we have tested it only on a reduced set of terms/concepts, it was difficult to compare its results with the results of *Text2Onto* and with the manually built ontology. However, we have identified a subset of common concepts between the concepts involved in the relations extracted by *WCL System*, the concepts involved in the relations extracted by *Text2Onto* and the concepts involved in taxonomic relations of the manually built ontology. A precise analysis of the relations between the concepts of this subset has underlined strong complementarities between the results of *WCL System* and *Text2Onto*, and between their results and the manually built ontology. As these tools implement very different techniques to extract taxonomic relations, these complementarities make sense to combine their results. In our experimentation on the subset of common concepts *WCL System* and *Text2Onto* have identified together 90% of the taxonomic relations that relate the 21 common concepts in the union ontology.

From a general point of view, the ontologies constructed by the analyzed tools contain instances, concepts and – in the best cases – taxonomic relations only. Despite the good precision of *TermExtractor*, the results are often incomplete but they can be used as a basis to help an expert in the ontology building process, or as additional resources to complete existing ontologies.

4 Conclusion

In this paper we have compared four approaches and related tools among the most common ones to automatically build ontologies from textual resources. Using

Methontology as a framework, we first have proposed a synthetic comparison of the tasks belonging to the four approaches. This synthetic comparison has highlighted a great variability of the used algorithms both in the concept extraction stages and in the relation establishment. Then, we have made an experimental comparison on corpus of about 4000 words.

Our analysis established a major difference between the approaches: some of them (Text2Onto, SPRAT) propose a fully automated ontology construction, while some others propose either separate tools for each task sometimes not fully integrated for ontology extraction (like OntoLearn) or just suggest tools taken from available prototypes and products (like Alvis).

From an experimental point of view, the tests confirm that tools can be used without any major problem on medium-size corpus. However, there are significant differences between those tools. Concerning concept-instance, *Text2Onto* has the best recall and *TermExtractor* the best precision, while *SPRAT* has a low recall. Concerning taxonomic relation, the relations identified by *WCL System* are semantically richer than the relation extracted by *Text2Onto*. Generally speaking, the obtained results allow us to say that *TermExtractor* + *WCL System* and *Text2Onto* seem to have a real potential for automating ontology construction. *SPRAT* is interesting but it needs text corpus which contains a significant number of expressions matching predefined patterns.

Tools automating ontology construction may speed up the ontology construction process (indeed they produce relevant concepts, instances and relationships) and this paper provides few elements for understanding how these tools can be selected and used for specific applications. However, several challenging points remain open. The first is about the relationship between tools performances (especially precision and recall), the implemented techniques and the input text features. A full development of this point should result in a decision support system or in a recommendation system supporting the selections. The second point concerns the integration between manual tasks/subtasks (Table 1) and fully automated tasks/subtasks. The third point is about the correction of automatically built ontologies: how errors can be identified, how ontologies can be improved and so on. Our further work is devoted to investigate these challenging points.

References

[Aime et al., 2009] Aime, X., Furst, F., Kuntz, P., Trichet, F.: Gradients de prototypicalité appliqués à la personnalisation d'ontologies. In: Actes de la Conférence Ingénierie des Connaissances (IC 2009), pp. 241–252 (2009)

[Bourigault and Lame, 2002] Bourigault, D., Lame, G.: Analyse distributionnelle et structuration de terminologie. application á la construction d'une ontologie documentaire du droit. Traitement Automatique des Langues 43(1), 129–150 (2002)

[Buitelaar et al., 2005] Buitelaar, P., Cimiano, P., Magnini, B.: Ontology learning from text: an overview. In: Ontology Learning from Text: Methods, Applications and Evaluation, pp. 3–12. IOS Press (2005)

[Cimiano and Volker, 2005] Cimiano, P., Völker, J.: Text2Onto - a Framework for Ontology Learning and Data-driven Change Discovery. In: Montoyo, A., Muńoz, R., Métais, E. (eds.) NLDB 2005. LNCS, vol. 3513, pp. 227–238. Springer, Heidelberg (2005)

[Corcho et al., 2005] Corcho, Ó., Fernández-López, M., Gómez-Pérez, A., López-Cima, A.: Building Legal Ontologies with METHONTOLOGY and WebODE. In: Benjamins, V.R., Casanovas, P., Breuker, J., Gangemi, A. (eds.) Law and the Semantic Web. LNCS (LNAI), vol. 3369, pp. 142–157. Springer, Heidelberg (2005)

[Fernandez et al., 1997] Fernández-López, M., Gómez-Pérez, A., Juristo, N.: Methontology: From ontological art towards ontological engineering. In: Proc. of the AAA 1997 Spring Symposium Series on Ontological Engineering, pp. 33–40 (1997)

[Gherasim et al., 2011] Gherasim, T., Harzallah, M., Berio, G., Kuntz, P.: Analyse comparative de méthodologies et d'outils de construction automatique d'ontologies á partir de ressources textuelles. In: EGC 2011 (2011)

[Gruber, 1993] Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquisition 5(2), 199–220 (1993)

[Harris, 1968] Harris, Z.: Mathematical Structures of Language. John Wiley and Son (1968)

[Hearst, 1992] Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th International Conference on Computational Linguistics, pp. 539–545 (1992)

[Maynard et al., 2009a] Maynard, D., Funk, A., Peters, W.: Nlp-based support for ontology lifecycle development. In: Proc. of ISWC Workshop on Collaborative Construction, Management and Linking of Ontologies (2009a)

[Maynard et al., 2009b] Maynard, D., Funk, A., Peters, W.: Sprat: a tool for automatic semantic pattern-based ontology population. In: Proc. of the Int. Conf. for Digital Libraries and the Semantic Web (2009b)

[Navigli and Velardi, 2004] Navigli, R., Velardi, P.: Learning domain ontologies from document warehouses and dedicated web sites. Computational Linguistics 30(2), 151–179 (2004)

[Navigli and Velardi, 2005] Navigli, R., Velardi, P.: Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 27(7), 1075–1086 (2005)

[Navigli et al., 2003] Navigli, R., Velardi, P., Gangemi, A.: Ontology learning and its application to automated terminology translation. IEEE Intelligent Systems 18(1), 22–31 (2003)

[Nazarenko and Hamon, 2002] Nazarenko, A., Hamon, T.: Structuration de terminologie: quels outils pour quelles pratiques? Traitement Automatique des Langues. Structuration de Terminologie 43(1), 7–18 (2002)

[Nedellec, 2006] Nedellec, C.: Semantic class learning and syntactic resources tuning. Technical report, Deliv. 6.4a for ALVIS (Superpeer semantic Search Engine) Project (2006)

[Osborne et al., 2009] Osborne, J., Flatow, J., Holko, M., Lin, S., Kibbe, W., Zhu, L., Danila, M., Feng, G., Chisholm, R.L.: Annotating the human genome with disease ontology. BMC Genomics 10(supl.1), 63–68 (2009)

[Park et al., 2011] Park, J., Cho, W., Rho, S.: Evaluating ontology extraction tools using a comprehensive evaluation framework. Data Knowl. Eng. 69, 1043–1061 (2011)

[Salton et al., 1975] Salton, G., Yang, C., Yu, C.: A theory of term importance in automatic text analysis. Journal of the American Society for Information Science 26, 33–34 (1975)

[Velardi et al., 2007] Velardi, P., Cucchiarelli, A., Pétit, M.: A taxonomy learning method and its application to characterize a scientific web community. IEEE Trans. on Knowl. and Data Eng. 19(2), 180–191 (2007)

[Velardi et al., 2008] Velardi, P., Navigli, R., D'Amadio, P.: Mining the web to create specialized glossaries. IEEE Intelligent Systems 23(5), 18–25 (2008)

[Volker and Sure, 2006] Volker, J., Sure, Y.: Data-driven change discovery - evaluation. Technical report, Deliv. D3.3.2 for SEKT Project, Instit. AIFB, Univ. of Karlsruhe, SEKT Deliv. (2006)

[Zouaq and Nkambou, 2010] Zouaq, A., Nkambou, R.: A Survey of Domain Ontology Engineering: Methods and Tools. In: Nkambou, R., Bourdeau, J., Mizoguchi, R. (eds.) Advances in Intelligent Tutoring Systems. SCI, vol. 308, pp. 103–119. Springer, Heidelberg (2010)

User Centered Cognitive Maps

Lionel Chauvin, David Genest, Aymeric Le Dorze, and Stéphane Loiseau

1 Introduction

Two kinds of influence graphs are commonly used in artificial intelligence to modelize influence networks: bayesian networks [Naïm et al., 2004] and cognitive maps [Tolman, 1948]. Influence graphs provide mechanisms to highlight the influence between concepts. *Cognitive maps* represent a *concept* by a text and an *influence* by an arc to which a value is associated. These values generally belong to sets of symbols like $\{+,-\}$ [Axelrod, 1976, Chauvin et al., 2008b], $\{none; some; much; a\ lot\}$ [Dickerson and Kosko, 1994, Zhou et al., 2003] or belong to sets of numeric values like $[-1,+1]$ [Kosko, 1986, Satur and Liu, 1999]. For symbolic set of values, a cognitive map can be represented as a conceptual graph[Baget and Mugnier, 2001] where concepts of a cognitive map are concepts of a conceptual graph and influences are particular relations. The difference is that cognitive map provides specific semantic for the influences. In cognitive map models the influences and their values are used in the computation of the *propagated influence* from a concept to another, according to the paths between these concepts. Cognitive maps have been used in many fields such as ecology [Celik et al., 2005], biology [Tolman, 1948], sociology [Poignonec, 2006], politics [Levi and Tetlock, 1980]. They are used to help a user to take a decision by understanding the consequences of it.

Cognitive maps have the drawback of not being so easy to exploit in practical applications. The main reason is the important number of concepts which makes cognitive maps difficult to construct and to apprehend. The main idea of this paper is to provide a solution to obtain views of a cognitive map adapted to what the user wants to do.

Our first contribution is to introduce the notion of scale in the cognitive map model, in order to let the user select the level of detail of the map he wants to

Lionel Chauvin · David Genest · Aymeric Le Dorze · Stéphane Loiseau
LERIA, UFR Sciences, 2 Bd Lavoisier, 49045 Angers Cedex 01, France
e-mail: {lionelc,genest,ledorze,loiseau}@info.univ-angers.fr

F. Guillet et al. (Eds.): *Advances in Knowledge Discovery & Management*, SCI 471, pp. 203–220.
DOI: 10.1007/978-3-642-35855-5_10

visualize. To do that, we associate an *ontology* to an initial cognitive map. The ontology is a taxonomy that organizes the concepts using a specialization relation. The most specialized concepts are called *elementary concepts*: these concepts are the only ones represented in the cognitive map. A *scale* is a subset of concepts of the ontology chosen by the user in order to provide a view for a cognitive map adapted to the user. Each elementary concept, or a concept that generalizes it, must belong to the scale. A view is a cognitive map computed using only the concepts of the scale: the view is then adapted to the user. It usually has fewer concepts than the elementary concepts, i.e. the concepts of the cognitive map; we speak of *view for a scale*.

Our second contribution is to automatically provide an adaptation of a cognitive map to a user in the form of an adapted view for him. To do that, a particular scale is associated to a user. We call such a scale a *profile*. When using a cognitive map, the profile associated to the user is used to compute a map, called a *view for a profile*, which is *adapted* to the user. When several users want to work on a single cognitive map, user profiles are combined together so as to construct a new scale composed of *shared concepts*. From the *shared concepts*, a *shared view* adapted to these users is computed.

In practice, to associate an ontology to a cognitive map, we define an *ontological cognitive map* (OCM) as the association of a cognitive map and an ontology whose elementary concepts are the concepts of the map. The *ontological influence* provides a way to compute the influence between any pair of concepts of the ontology. The ontological influence is used to compute the value of the influences in a view. Note that an ontology has already been associated to cognitive maps, for instance in [Jung et al., 2003] and in [Poignonec, 2006] where it is used to compare or to merge maps.

The second section of this paper presents related works. The third section describes the OCM model and the propagated influence. The fourth section introduces the notion of scale and view for a scale. The fifth section defines the notions of profile, view for a profile and shared view. The sixth section introduces different ways to compute the propagated influence according to the set of values associated to the map.

2 Related Works

In this section we first recall what the categories of support systems are. Second, we show how graphical knowledge models, especially cognitive maps, can be considered in the category of decision support systems. Third, we present three major approaches used by cognitive maps. Fourth, we present an approach mixing different graphical knowledge models, i.e. cognitive maps and conceptual graphs. Fifth, we discuss the concept of ambiguities in cognitive maps.

Support systems in business are typically classified in 3 categories: *EIS* (Executive Information Systems), *ESS* (Executive Support Systems) and *DSS* (Decision Support Systems) [Turban, 1993]. EIS focuses on the construction of synthetic data, mostly from databases, highlighting the data that seem most relevant in relation to the chosen objective. For instance, [Paradice, 1992] proposes a hybrid model as the combination of an object model and a causal model. ESS integrate information from previous systems but operate more by providing a prospective case study or simulation of several scenarios. For instance, [Vasan, 2003] proposes a system of multiple simulations using fuzzy linear programming. The DSS are systems that strongly interact with users: they use different decision models, using data that may be poorly structured. They are used for complex problems and provide mechanisms for cooperative work. For instance, [Pinson et al., 1997] proposes a distributed decision system for strategic planning and [Rommelfanger, 2004] presents a review of fuzzy optimization models for decision support.

Graphical knowledge models, especially cognitive maps, have been defined. Most of these models can be considered as decision support systems. For these models, a decision is often a choice of a user among different alternatives to reach the goal that he fixes. The models help a user to express knowledge about these alternatives, and help him to take his decision. Representation of these alternatives helps the user to take into account the links existing between events or notions: by knowing the consequences of his choice, a user can take his decision. However, to make this help efficient, the user must understand the represented knowledge, he must access easily to the alternatives and he must find and understand their consequences. Several models provide graphical representations of alternatives and links between concepts, some of these models are based on the notion of a map such as mind maps [Buzan and Buzan, 2003], concept maps [Novak and Gowin, 1984] and cognitive maps [Axelrod, 1976]. Among these models, cognitive maps are one of the easiest to use.

Currently, cognitive maps are used primarily in 3 major approaches. First, cognitive maps are used to assist in the structuring of thought before taking a decision [Huff and Fiol, 1992]. A cognitive map can clarify a confused idea because it models representations and it acts on this representation in the structuring process, the cognitive map metaphor is thus used to model the biological role of the hippocampus [Redish, 1999]. Second, cognitive maps are used as a medium for communication about a decision between individuals [Eden, 1988] or agents [Chaib-draa, 2002, Tisseau, 2001] [Parenthoen et al., 2001]. The development of a cognitive map facilitates the transmission of ideas between decision makers and becomes a communication tool. Third, cognitive maps are used to make decisions [Huff and Fiol, 1992][Ronarc'h et al., 2005]. The cognitive map is a model designed to include the path by which an individual will find a solution to a given problem: this path may be computed automatically.

[Genest and Loiseau, 2007] and [Chauvin et al., 2008a] proposes an extended model of cognitive maps that mix the cognitive map model with a conceptual graph model. First, concepts are expressed with a conceptual graph[Sowa, 1984]. It provides clear *definitions of concepts*. These definitions are taken into account to provide an efficient search mechanism: a user may build a query using a conceptual graph, and the system can extract the concepts of the map that correspond to the query. So, *sets of concepts* can be automatically built, and this paper proposes some specific inference mechanisms on sets of concepts. Second, a *validity context* is given for each influence of a map in the form of a conceptual graph. The validity context of an influence represents cases in which this influence is relevant. For each category of user, a *use context* is defined using a conceptual graph. Using validity contexts, a *filtering mechanism* extracts concepts and influences that are relevant for one context. So for a user and his use context, the resulting cognitive map is simpler than the initial cognitive map and makes possible the computation of propagated influences that fit him better. Since conceptual graphs are graphs, conceptual graph model is homogenous with the visual aim of cognitive maps. The conceptual graph model defines operations and has a logical semantics. A conceptual graph is a graph composed of concept nodes representing entities and relation nodes representing relations between entities. A graph is defined on a structure called support that specifies and organizes in a hierarchy the basic vocabulary used for concepts and relations. The support is an ontology of the domain. A formal operation, called projection, provides a way to search logical links between graphs and is used as a base of the search mechanism and the filtering mechanism. Some works on conceptual graphs are intended to facilitate the knowledge modeling from several experts. These include for example the model C-Vista [Ribière and Dieng-Kuntz, 2002].

The preceding approaches using cognitive maps and conceptual graphs removes the ambiguous results of the influence propagation by taking into account only the influences relevant to a user profile. Other works address the same problem but solve the ambiguities using different sets of influence values associated to an operation of influence propagation based on logics. [Zhang et al., 1992] [Zhang, 1996] propose solutions for cognitive maps they describe as vague in defining the opportunity to obtain two different views on influence, for example, one positive and another very negative. The need for such solutions depends on the application, it should be noted that the use of links that are not symbolic elements restrict the use of cognitive maps to expert users. Truck's thesis [Truck, 2002] offers a state of the art solutions to aggregate and make inferences with different operators and is a possible entry point for thinking about the design of extensions of cognitive maps incorporating links or fuzzy data.

3 Ontological Cognitive Map and Inference

An ontological cognitive map (OCM) is the association of a cognitive map and an ontology. The propagated influence between two concepts is a value computed using the influence paths from one concept to the other. The ontological influence is a generalization of the propagated influence to every ordered pair of concepts of the ontology.

A cognitive map is an oriented graph where nodes are labeled by concepts. A concept is a text. An arc is labeled by a value that describes the effect of the influence.

Definition 1 (Cognitive map). Let I be a set of values. Let C be a set of concepts. A *cognitive map* defined on C and I, is an oriented labeled graph $(V, label_V, A, label_A)$ where:

- V is a set of nodes.
- $label_V : V \rightarrow C$ is a bijective function labeling a node of V with a concept of C
- $A \subseteq V \times V$ is a set of arcs called *influences*
- $label_A : A \rightarrow I$ is a function labeling an influence with a symbol of I.

Example 1. *Map1* (figure 1) concerns a road safety analysis. *Map1* is defined on $I = \{+, -\}$ and represents the influence of different factors on the risk that an accident occurs. For instance, driving in the rain positively influences the risk of the road to being slippery, so there is an arc labeled by a $+$ symbol between the concepts *Rain* and *Slippery road*. On the contrary, *Motorway* negatively influences *Winding road*, so there is an arc labeled by a $-$ symbol.

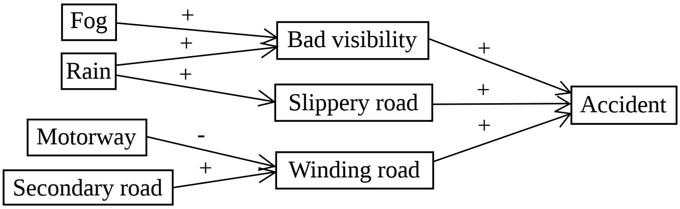


Fig. 1 *Map*1: a cognitive map about road safety problems

An ontology in an OCM is represented by a set of concepts partially ordered by a specialization relation. For a subset of concepts of an ontology, minimum (resp. maximum) concepts are the concepts for which there are no lesser (resp. greater) concepts than them. An ontological cognitive map is the association of a cognitive map and an ontology. Only the elementary concepts of the ontology are represented in the map because influences are defined only on them.

Definition 2 (Ontology). An *ontology* $(C, \preceq)$ is a set of concepts C partially ordered by a relation $\preceq$. We note $\prec$ the strict order relation associated with $\preceq$.

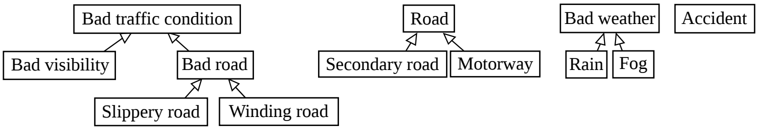


Fig. 2 Ontology1

Definition 3 (Maximum, minimum and elementary concepts). Let $(C, \preceq)$ be an ontology. Let $C' \subseteq C$. We name the set of *maximum concepts* of C': $max(C') = \{c \in$

$C' \mid \nexists\, c' \in C' \,,\ c \prec c'\}$. We name the set of *minimum concepts* of C': $min(C') = \{c \in C' \mid \nexists\, c' \in C' \,,\ c' \prec c\}$. The concepts of $min(C)$ are called *elementary concepts*.

Definition 4 (OCM). An *ontological cognitive map* defined on an ontology $(C, \preceq)$ and a set of values I is an association of an ontology $(C, \preceq)$ and a cognitive map defined on $min(C)$ and I.

Example 2. *OMAP1* is the OCM built by associating the ontology *Ontology1* (figure 2) and the cognitive map *Map1*. *Motorway* $\preceq$ *Road* means that a motorway is a kind of road. Note that *Map1* only contains the elementary concepts of *Ontology1*. $min(Ontology1) = \{\mathit{Bad\ visibility}, \mathit{Slippery\ road}, \mathit{Winding\ road}, \mathit{Rain}, \mathit{Fog}, \mathit{Secondary\ road}, \mathit{Motorway}, \mathit{Accident}\}$

The propagated influence of a concept on another is computed according to the influence paths existing between the nodes labeled by these concepts.The propagated influence for an influence path is evaluated by cumulating all values of its influences. Definitions 6,7,9 of operators are made for the set of values $I = \{+, -\}$. Section 5 discusses how to adapt these definitions for other sets of values.

Definition 5 (Influence path). Let $M = (V, label_V, A, label_A)$ be a cognitive map defined on a set of concepts C and a set of values I. Let c_1, c_2 be two concepts of C.

- We name an *influence path* from c_1 to c_2 a sequence (of length k) of influence $(u_i, v_i) \in A$ such that $u_1 = label_V^{-1}(c_1)$ and $v_k = label_V^{-1}(c_2)$ and $\forall i \in [1..k-1], v_i = u_{i+1}$.
- An influence path P from the concept c_1 to c_2 is *minimal* iff an influence path P' from c_1 to c_2 such that P' is a subsequence of P does not exist.
- We note $\mathcal{P}_{c_1,c_2}$ the *set of minimal influence paths* from c_1 to c_2.

Definition 6 (Propagated influence for an influence path). Let $M = (V,\ label_V,\ A,\ label_A)$ be a cognitive map defined on a concept set C and the set of influence values $I = \{+, -\}$.

The *propagated influence for an influence path P* is:

$$\mathcal{I}_P(P) = \bigwedge_{(v,v')\ of\ P} label_A((v, v'))$$

with $\bigwedge$ a function defined on $I \times I \to I$ and represented by the table 1.

Table 1 $\bigwedge$ operator

$\bigwedge$	+	-
+	+	-
-	-	+

The propagated influence from a concept to another concept can be null (noted by 0) if no path exists between these concepts. It is positive when the influences propagated in all the paths between these concepts are positive (noted by +). It is negative when the influences propagated in all the paths between these concepts are negative (noted by -). When two or more paths have different propagated influences, it is not possible to decide if the propagated influence between these two concepts is positive or negative. It is also not possible to know if the paths compensate each others (in this case it would be null). In such a case, the propagated influence is ambiguous (noted by ?). This mechanism has the drawback to often return ambiguous results.

Definition 7 (Propagated influence). Let $M = (V, label_V, A, label_A)$ be a cognitive map defined on a concept set C and the set of influence values $I = \{+, -\}$.

The *propagated influence* between two concepts is a function $\mathcal{I}$ defined on $C \times C \to \{0, +, -, ?\}$ such that:

$$\mathcal{I}(c_1, c_2) = 0 \; if \;\; \mathcal{P}_{c_1,c_2} = \emptyset$$

$$\mathcal{I}(c_1, c_2) = \bigvee_{P \in \mathcal{P}_{c_1,c_2}} \mathcal{I}_P(P) \; if \;\; \mathcal{P}_{c_1,c_2} \neq \emptyset$$

where $\bigvee$ is a function defined on $\{+, -, ?\} \times \{+, -, ?\} \to \{+, -, ?\}$ represented by the table 2.

Table 2 $\bigvee$ operator

$\bigvee$	+	-	?
+	+	?	?
-	?	-	?
?	?	?	?

Example 3. We want to compute the influence between `Rain` and `Accident`. Two influence paths are presented in $Map1$ between these concepts: p_1 (*Rain* → *Bad visibility* → *Accident*) and p_2 (*Rain* → *Slippery road* → *Accident*).

$\mathcal{I}(\mathit{Rain}, \mathit{Accident}) = (\mathcal{I}_P(p_1) \vee \mathcal{I}_P(p_2)) = ((+ \wedge +) \vee (+ \wedge +)) = +$.

The ontological influence provides a mechanism to the user that queries an OCM to determine the influence between any ordered pair of concepts of the ontology. For this, we first determine the two subsets of elementary concepts that specialize the two concepts of the pair. The ontological influence between two concepts c_1 and c_2 is then the aggregation of values of the influences propagated between the elementary concepts of c_1 and those of c_2.

We propose to add two symbols $\oplus$ and $\ominus$. The first symbol represents the value of an influence that is positive or null. The second symbol represents the value of an influence that is negative or null. These new symbols simplify the reading of the ontological influence.

Definition 8 (Elementary concepts for a concept). Let $(C,\preceq)$ be an ontology. Let c be a concept of C. We name the set of *elementary concepts for a concept c*, the subset of C defined as:

$$elemFor(c) = \{c' \in min(C) | c' \preceq c\}$$

Definition 9 (Ontological influence). Let *OM* be an ontological cognitive map defined on an ontology $(C,\preceq)$ and the set of influence values $\{+,-\}$.

The *ontological influence* between two concepts of C is a function $\mathcal{I}_O$ defined on $C \times C \rightarrow \{+,-,0,?\}$ such that:

$$\mathcal{I}_O(c_1,c_2) = \bigodot_{\substack{c'_1 \in elemFor(c_1) \\ c'_2 \in elemFor(c_2)}} \mathcal{I}(c'_1,c'_2)$$

where $\bigodot$ is a function defined on $\{+,-,0,?\} \times \{+,-,0,?\} \rightarrow \{+,-,0,?\}$ represented by the table 3.

Table 3 $\odot$ operator

$\odot$	+	-	0	$\oplus$	$\ominus$	?
+	+	?	$\oplus$	$\oplus$	?	?
-	?	-	$\ominus$	?	$\ominus$	?
0	$\oplus$	$\ominus$	0	$\oplus$	$\ominus$	?
$\oplus$	$\oplus$	?	$\oplus$	$\oplus$	?	?
$\ominus$	?	$\ominus$	$\ominus$	?	$\ominus$	?
?	?	?	?	?	?	?

Example 4. We want to compute the ontological influence between the concept *Bad weather* and the concept *Bad traffic condition*.

First, we determine the elementary concepts for *Bad weather* and for *Bad traffic condition*:

- $elemFor$(*Bad weather*) $= \{$*Fog*, *Rain*$\}$
- $elemFor$(*Bad traffic condition*) $= \{$*Bad visibility*, *Slippery road*, *Winding road*$\}$

Second, we compute the influence between each possible ordered pair (c_1,c_2) where c_1 is a member of $elemFor$(*Bad weather*) and c_2 is a member of $elemFor$(*Bad traffic condition*):

- $\mathcal{I}$(*Fog*, *Bad visibility*) $= +$.
- $\mathcal{I}$(*Fog*, *Slippery road*) $= 0$.
- $\mathcal{I}$(*Fog*, *Winding road*) $= 0$.
- $\mathcal{I}$(*Rain*, *Bad visibility*) $= +$.
- $\mathcal{I}$(*Rain*, *Slippery road*) $= +$.
- $\mathcal{I}$(*Rain*, *Winding road*) $= 0$.

Third, we agregate the previous propagated influences using the operator $\odot$: $\mathcal{I}_O(Bad$ *weather*, *Bad traffic condition*$) = + \odot 0 \odot 0 \odot + \odot + \odot 0 = \oplus$.

The ontological influence between *Bad weather* and *Bad traffic condition* is positive or null.

4 View for a Scale

A scale is a subset of concepts from the ontology, chosen by the user in order to obtain a view. The concepts of the scale will be present in the view.

A scale respects some particular properties: all the concepts must be incomparable and they must be representative of all the concepts of the ontology.

Intuitively, the incomparability avoids taking into account twice the same concept in the scale: once as a concept and once as a concept that generalizes it. Intuitively, the representative ensures that every elementary concept is represented in the scale or a concept that generalizes it.

Definition 10 (Comparable concepts). Let $(C, \preceq)$ be an ontology. Two concepts c and c' of C are *comparable* iff $c \preceq c'$ *or* $c' \preceq c$.

Property 1 (Set of incomparable concepts). *Let* $(C, \preceq)$ *be an ontology. Let* $C' \subseteq C$. C' *is a* set of incomparable concepts *iff* $\forall c, c' \in C'$ *with* $c \neq c'$, c *and* c' *are not comparable.*

Definition 11 (Elementary concepts for a set). Let $(C, \preceq)$ be an ontology. Let $C' \subseteq C$. We name the *set of elementary concepts for a set* C': $elemForSet(C') = \bigcup_{c \in C'} elemFor(c)$.

Property 2 (Representative set of a set). *Let* $(C, \preceq)$ *be an ontology. Let* $C_1, C_2 \subseteq C$. C_1 *is* a representive set *of* C_2 *iff* $elemForSet(C_2) \subseteq elemForSet(C_1)$.

Theorem 1 shows that a set is representative of the ontology if its elementary concepts are the elementary concepts of the ontology.

Theorem 1. *Let* $(C, \preceq)$ *be an ontology. Let* $C' \subseteq C$. C' *is a representative set of* C *iff* $elemForSet(C') = min(C)$.

Definition 12 (Scale). Let $(C, \preceq)$ be an ontology. Let $C' \subseteq C$. C' is a *scale* iff C' **1)** is a set of incomparable concepts (Property 1) and **2)** is representative of C (Property 2).

Example 5. Let $A = \{$*Bad traffic condition*, *Bad weather*, *Road*, *Accident*$\}$. A respects property 1 because *Bad traffic condition*, *Bad weather*, *Road*, *Accident* are not comparable. A respects property 2 because $elemForSet(A) = \{$*Bad visibility*, *Slippery road*, *Winding road*, *Rain*, *Fog*, *Secondary road*, *Motorway*, *Accident*$\} = min(Ontology1)$. So, A is a scale.

A view of an OCM is a cognitive map in which concepts are those of a scale. Two concepts of a view are connected if there is one elementary concept for each of them so that those two elementary concepts are connected in the OCM. An arc between two elementary concepts of the view is labeled in the same way as the corresponding arc of the OCM. In other cases, the value of an arc in the view is computed using the ontological influence.

Definition 13 (Connection between two concepts)
Let $OM = (V, label_V, A, label_A)$ be an OCM defined on an ontology $(C, \preceq)$ and a set of values I. Two concepts c_1 and c_2 of C are *connected* iff $\exists c'_1 \in elemFor(c_1)$, $\exists c'_2 \in elemFor(c_2) \mid (label_V^{-1}(c_1), label_V^{-1}(c_2)) \in A$.

Definition 14 (Value of an influence between two connected concepts). Let $OM = (V, label_V, A, label_A)$ be an OCM defined on an ontology $(C, \preceq)$ and a set of values I. $\forall c_1, c_2 \in C$ that are connected:

$$Value(c_1, c_2) = \begin{cases} label_A(label_V^{-1}(c_1),\ label_V^{-1}(c_2))\ if \\ c_1\ and\ c_2\ are\ elementary\ concepts. \\ \mathcal{I}_O(c_1, c_2)\ otherwise. \end{cases}$$

Definition 15 (View for a scale). Let $OM = (V, label_V, A, label_A)$ be an OCM defined on an ontology $(C, \preceq)$ and a set of values I. Let C' be a scale. A *view for* C' of OM is a cognitive map $(V_s, label_{V_s}, A_s, label_{A_s})$ defined on C' and I such that:

- V_s is a set of node whose cardinality is equal to the cardinality of C'.
- $label_{Vs} : V_s \to C'$ is a bijective function labeling each node of V_s with a concept of C'.
- $A_s \subseteq V_s \times V_s$ is the set of influences $(label_{Vs}^{-1}(c_1), label_{Vs}^{-1}(c_2))$ such that c_1 and c_2 are connected.
- $label_{A_s} : A_s \to I \cup I \times I$ is a labeling function such that $label_{A_s}((v_1, v_2)) = Value(label_{V_s}(v_1), label_{V_s}(v_2))$

Example 6. Figure 3 is the view of *OMAP1* for the scale {*Bad traffic condition*, *Road*, *Bad weather*, *Accident*}. The color gray of boxes are the maximum concepts introduced by the scale. We note that the influence between *Bad weather* and *Bad traffic condition* is labeled by $\oplus$ as seen in the example 4. The influence between *Road* and *Bad traffic condition* is labeled by ? because, in *Map1*, there is a positive influence between *Motorway* and *Winding road* and there is a negative influence between *Secondary road* and *Winding road*. The influence between *Bad traffic condition* and *Accident* is labeled by + because all influences from an element of *elemFor*(*Bad traffic condition*) to *Accident* in *Map1* are positive.

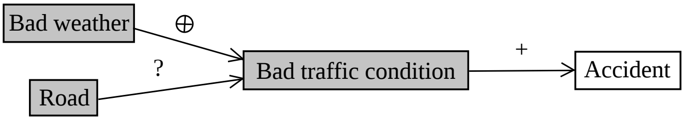


Fig. 3 View of *OMAP*1

5 Shared View

To a user is associated a profile that defines a scale that fits to him. This profile provides a solution to obtain a view well adapted to the user: it is the *view for the profile*.

Definition 16 (User profile). Let $(C, \preceq)$ be an ontology. A *user profile* is a scale for C.

Definition 17 (View for a profile). Let OM be an OCM defined on $(C, \preceq)$ and I. Let P be a user profile. The *view for a profile P* is the view for P of OM.

Example 7. Figure 4 presents the view for the profile $P_m = \{$*Fog, Rain, Road, Bad traffic condition, Accident*$\}$ built for the user "meteorologist". Another view for the profile $P_r = \{$*Motorway, Secondary road, Bad weather, Bad traffic condition, Accident* $\}$ can be computed for the user "road constructor".

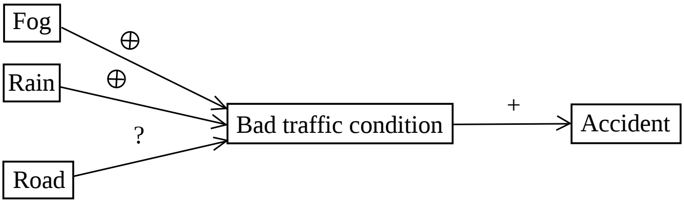


Fig. 4 View for the user "meteorologist"

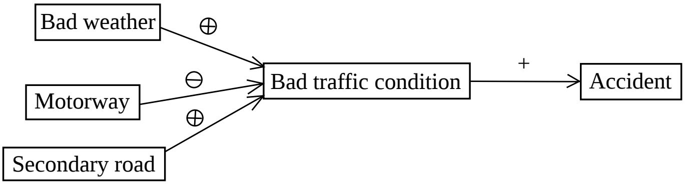


Fig. 5 View for the user "road constructor"

When two users share the same map and want to use it together, a shared view, adapted to the two users, will be built from a scale compound of all the concepts shared by two users. This set of shared concepts is the union of the two user profiles to which a min is applied for two reasons. First to provide the most specialized concepts relevant to both users, second to ensure that all shared concepts is a scale.

Definition 18 (Shared concepts). Let OM be an OCM defined on $(C, \preceq)$ and I. Let P_1 and P_2 be two profiles. $SharedConcepts(P_1, P_2) = min(P_1 \cup P_2)$.

Property 3 enables to use the shared concepts in order to compute a view.

Property 3 . *Let OM be an OCM defined on $(C, \preceq)$ and I. Let P_1 and P_2 be two profiles. $SharedConcepts(P_1, P_2)$ is a scale for C.*

Definition 19 (Shared view). Let OM be an OCM defined on $(C, \preceq)$ and I. Let P_1 and P_2 be two profiles. The *shared view* for P_1 and P_2 is the view for $SharedConcepts(P_1, P_2)$ of OM.

Example 8. Figure 6 presents the shared view for the two profiles P_r and P_m. For a better presentation, the concepts which are in P_r are represented by boxes whose borders are small dashes; the concepts of P_m are represented in gray boxes.

The user "meteorologist" finds in this view the concepts that interest him particularly: *Fog* and *Rain*. The user "road constructor" finds in this view the concepts that interest him particularly: *Motorway* and *Secondary road*. Using this view they can talk about the influence of their particular interest on the *Bad traffic condition* or *Accident*.

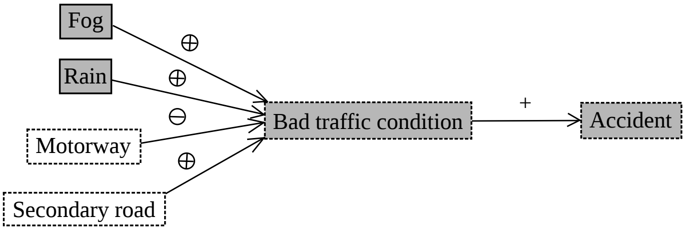


Fig. 6 Shared view for P_r and P_m

Shared view can be trivially generalized to more than two users.

6 Parameters

In the previous sections, the cognitive maps have been defined on the set of values $I = \{+, -\}$, and operators have been defined for this set. It is possible to change the operators used in the definitions 6,7,9 for a new set of values.

For $I = [-1, +1]$ the propagated influence for an influence path and the propagated influence between two concepts are given in definition 20, in conformity with [Kosko, 1986]. The ontological influence is given in definition 20 in conformity with [Chauvin et al., 2008b].

Definition 20 (Propagated influence ($I = [-1,+1]$)). Let $(V, label_V, A, label_A)$ an ontological cognitive map defined on the ontology $(C, \preceq)$ and on the set of values $I = [-1,+1]$.

- The *propagated influence for an influence path P* is:

$$\mathcal{I}_P(P) = \prod_{(v,v')\ of\ P} label_A((v,v'))$$

- The *propagated influence between two concepts* is a function $\mathcal{I}$ defined on $C \times C \rightarrow I$ such that:

$$\mathcal{I}(c_1,c_2) = \begin{cases} \frac{\sum_{P \in \mathcal{P}_{c_1,c_2}} \mathcal{I}_P(P)}{card(\mathcal{P}_{c_1,c_2})} & if\ \mathcal{P}_{c_1,c_2} \neq \emptyset \\ 0 & otherwise. \end{cases}$$

- The *ontological influence between two concepts* c_1, c_2 *of* C is a function $\mathcal{I}_O$ defined on $C \times C \rightarrow I \times I$ such that:

$$\mathcal{I}_O(c_1,c_2) = [\min_{\substack{c'_1 \in elemFor(c_1) \\ c'_2 \in elemFor(c_2)}} \mathcal{I}(c'_1,c'_2), \max_{\substack{c'_1 \in elemFor(c_1) \\ c'_2 \in elemFor(c_2)}} \mathcal{I}(c'_1,c'_2)]$$

Example 9. Let *OCM2* be an ontological cognitive map based on *OCM1* but labeled by the set of values $I = [-1,+1]$. The figure 7 represents *OCM2* and the view of *OCM2* for the profile "meteorologist".

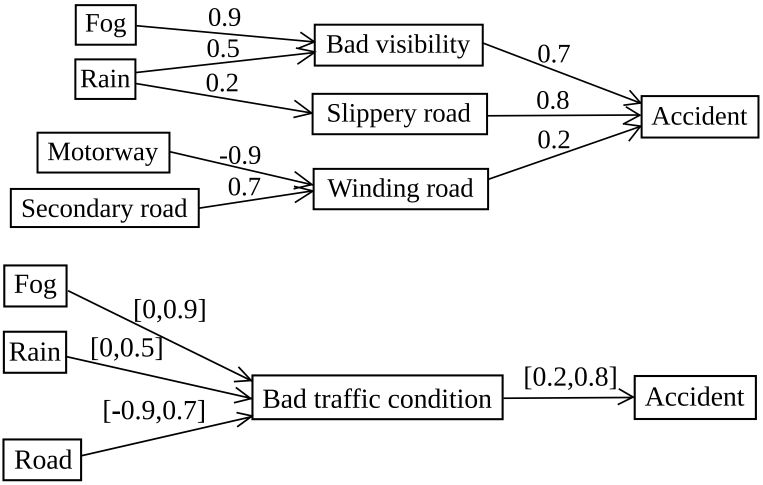


Fig. 7 *OCM2* and the view of *OCM2* for the profile "meteorologist"

For $I = \{null, some, much, a\ lot\}$, the propagated influence for an influence path and the propagated influence between two concepts are given in definition 21, in conformity with [Zhou et al., 2003].

The ontological influence is the same as the one of the definition 20. It returns an interval between two values of I.

Definition 21 (Propagated influence ($I = \{$*null* $\preceq$ *some* $\preceq$ *much* $\preceq$ *a lot*$\}$)). Let $(V, label_V, A, label_A)$ an ontological cognitive map defined on the ontology $(C, \preceq)$ and on the partial ordered set $I = \{null \preceq some \preceq much \preceq a\ lot\}$.

- The *propagated influence in a path P* is defined such as:

$$\mathcal{I}_P(P) = \min_{(v,v') \text{ of } P} label_A((v,v'))$$

- The *propagated influence between two concepts* c_1 and c_2 is defined such as:

$$\mathcal{I}(c_1,c_2) = \begin{cases} null & \text{if } \mathcal{P}_{c_1,c_2} = \emptyset \\ \max\limits_{P \in \mathcal{P}_{c_1,c_2}} \mathcal{I}_P(P) & \text{if } \mathcal{P}_{c_1,c_2} \neq \emptyset \end{cases}$$

- The ontological influence between two concepts is defined in one of definition 20.

Example 10. Let *OCM3* be an ontological cognitive map based on *OCM1* but labeled by the set of values $I = \{null \preceq some \preceq much \preceq a\ lot\}$. The figure 8 represents *OCM3* and the view for the profile "meteorologist".

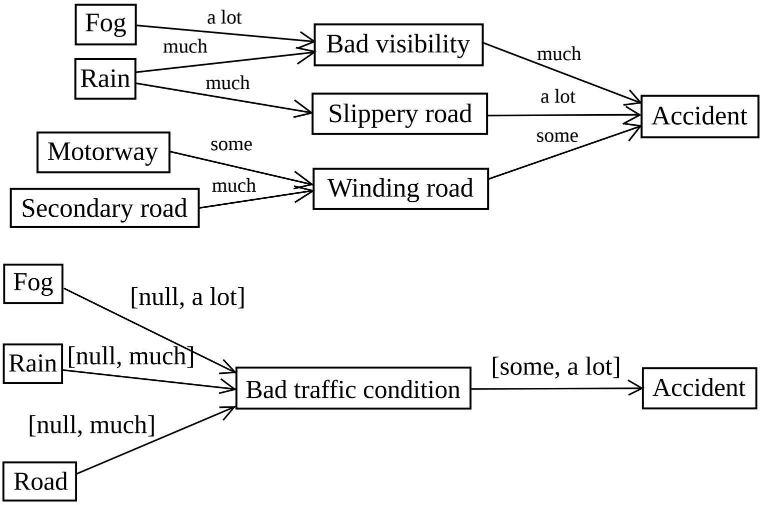


Fig. 8 *OCM3* and the view of *OCM3* for the profile "meteorologist"

Notice that other sets of values and operators can be proposed.

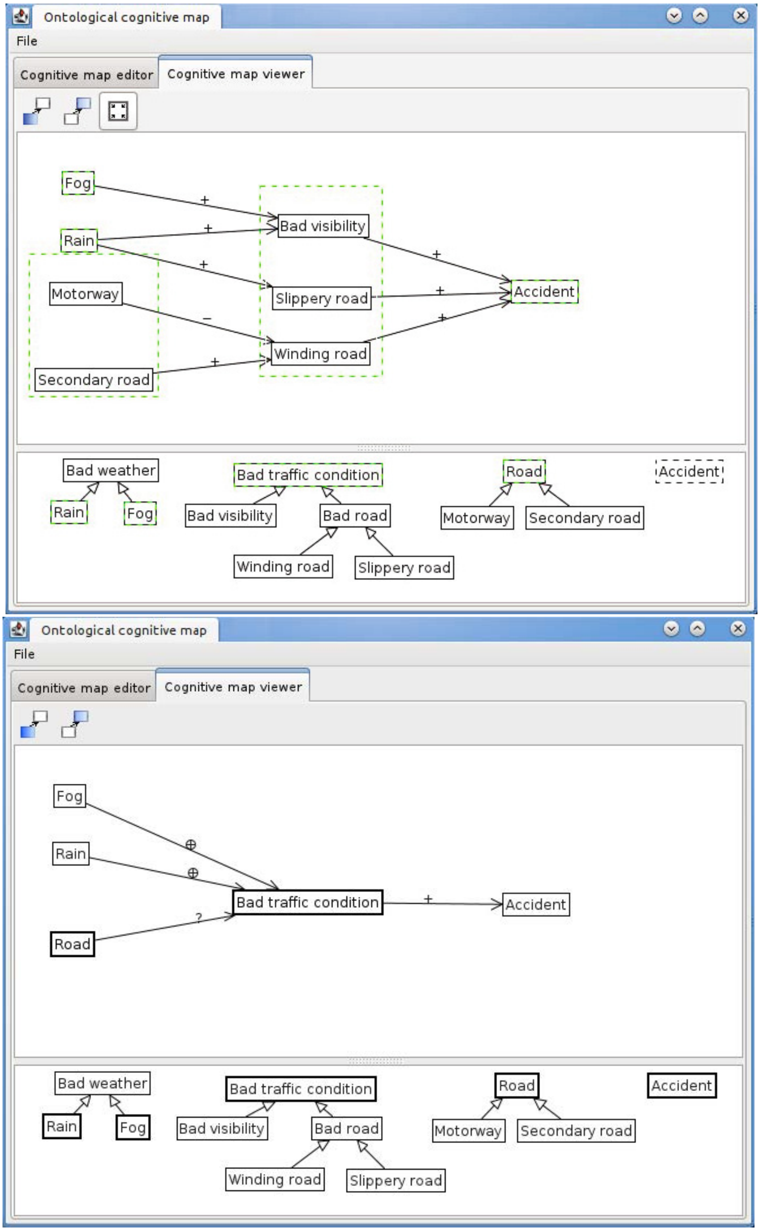


Fig. 9 Prototype: selection of a scale and a view of a cognitive map

7 Conclusion

This work extends the model of cognitive maps and its associated reasoning mechanisms in order to organize the concepts. It provides synthetic views of a cognitive map, using scales to see maps of different conceptual levels. It also proposes to take into account the user to adapt maps to him or her. The idea behind these extensions is that both during its edition and during its use, it is important to "navigate" in the map space so as to have different points of view; information considered important from a certain point of view is not the same as those that considered important from another. This idea of navigation in a cognitive map is new: previous works on cognitive maps are usually interested in the edition of maps, in the computation of propagation between concepts, or the comparison of maps.

SCCO (figure 9)[1] is a prototype that implements the ideas of this article. SCCO is able to add an ontology to cognitive maps; it provides mechanisms to compute the ontological influence of a concept of the ontology to another concept; it defines a scale as a subset of an ontology checking representativeness and incomparability properties; it uses a scale to produce a suitable view; it also proposes a profile and a shared view. Ontological cognitive maps containing about fifty concepts have been built with this program. It shows the interest of our approach.

References

[Axelrod, 1976] Axelrod, R.: Structure of decision: the cognitive maps of political elites, Princeton, N.J (1976)

[Baget and Mugnier, 2001] Baget, J.-F., Mugnier, M.-L.: The SG Family: extensions of simple conceptual graphs. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence, pp. 205–210. Morgan Kaufmann Publishers (2001)

[Buzan and Buzan, 2003] Buzan, T., Buzan, B.: The Mind Map Book. BBC Active (2003)

[Celik et al., 2005] Celik, F.D., Ozesmi, U., Akdogan, A.: Participatory ecosystem management planning at tuzla lake (turkey) using fuzzy cognitive mapping (2005)

[Chaib-draa, 2002] Chaib-draa, B.: Causal Maps: Theory, Implementation and Practical Applications in Multiagent Environments. IEEE Transactions on Knowledge and Data Engineering 14(2), 1–17 (2002)

[Chauvin et al., 2008a] Chauvin, L., Genest, D., Loiseau, S.: Contextual Cognitive Map. In: Eklund, P., Haemmerlé, O. (eds.) ICCS 2008. LNCS (LNAI), vol. 5113, pp. 231–241. Springer, Heidelberg (2008a)

[Chauvin et al., 2008b] Chauvin, L., Genest, D., Loiseau, S.: Ontological cognitive map. In: ICTAI 2008, vol. 2(1), pp. 225–232 (2008b)

[Dickerson and Kosko, 1994] Dickerson, J.A., Kosko, B.: Virtual worlds as fuzzy cognitive maps. Presence 3(2), 73–89 (1994)

[Eden, 1988] Eden, C.: Cognitive mapping. European Journal of Operational Research 36, 1–13 (1988)

[1] http://forge.info.univ-angers.fr/~lionelc/CCdeGCjava/

[Genest and Loiseau, 2007] Genest, D., Loiseau, S.: Modélisation, classification et propagation dans des réseaux d'influence. Technique et Science Informatiques 26(3-4), 471–496 (2007)

[Huff and Fiol, 1992] Huff, A.S., Fiol, M.: Maps for managers: where are we? where do we go from here? Journal of Management Studies 29, 267–285 (1992)

[Jung et al., 2003] Jung, J.J., Jung, K.-Y., Jo, G.-S.: Ontological Cognitive Map for Sharing Knowledge between Heterogeneous Businesses. In: Yazıcı, A., Şener, C. (eds.) ISCIS 2003. LNCS, vol. 2869, pp. 91–98. Springer, Heidelberg (2003)

[Kosko, 1986] Kosko, B.: Fuzzy cognitive maps. International Journal of Man-Machines Studies 24, 65–75 (1986)

[Levi and Tetlock, 1980] Levi, A., Tetlock, P.E.: A cognitive analysis of japan's 1941 decision for war. The Journal of Conflict Resolution 24(2), 195–211 (1980)

[Naïm et al., 2004] Naïm, P., Wuillemin, P.-H., Leray, P., Pourret, O., Becker, A.: Réseaux bayésiens. Eyrolles, Paris (2004)

[Novak and Gowin, 1984] Novak, J.D., Gowin, D.B.: Learning how to learn. Cambridge University Press, New York (1984)

[Paradice, 1992] Paradice, D.: Simon: an object oriented information system for coordinating strategies and operations. IEEE Transaction on System, Man and Cybernetics 22(3), 513–525 (1992)

[Parenthoen et al., 2001] Parenthoen, M., Tisseau, J., Reignier, P., Dory, F.: Agent's perception and charactors in virtual worlds: put fuzzy cognitive maps to work. In: Proceedings VRIC, pp. 11–18 (2001)

[Pinson et al., 1997] Pinson, S., Anacleto, J., Moraitis, P.: A distributed decision support system for strategic planning. International Journal of Decision Support Systems 20(1), 35–51 (1997)

[Poignonec, 2006] Poignonec, D.: Apport de la combinaison cartographie cognitive/ontologie dans la compréhension de la perception du fonctionnement d'un écosystème récifo-lagonaire de Nouvelle-Calédonie par les acteurs locaux. PhD thesis, ENSA Rennes France (2006)

[Redish, 1999] Redish, A.D.: Beyond the Cognitive Map: From Place Cells to Episodic Memory. MIT Press (1999)

[Ribière and Dieng-Kuntz, 2002] Ribière, M., Dieng-Kuntz, R.: A Viewpoint Model for Cooperative Building of an Ontology. In: Priss, U., Corbett, D.R., Angelova, G. (eds.) ICCS 2002. LNCS (LNAI), vol. 2393, pp. 220–234. Springer, Heidelberg (2002)

[Rommelfanger, 2004] Rommelfanger, J.: The advantages of fuzzy optimization models in practical use. Fuzzy Optimization and Decision Making 3, 295–309 (2004)

[Ronarc'h et al., 2005] Ronarc'h, N., Rozec, G., Guillet, F., Nédélec, A., Baquedano, S., Philippé, V.: Modélisation des connaissances émotionnelles par les cartes cognitives floues. In: Atelier Modélisation Conférence EGC, pp. 11–21 (2005)

[Satur and Liu, 1999] Satur, R., Liu, Z.-Q.: A contextual fuzzy cognitive map framework for geographic information systems. IEEE Transactions on Fuzzy Systems 7(5), 481–494 (1999)

[Sowa, 1984] Sowa, J.F.: Conceptual Structures: Information Processing in Mind and Machine. Addison Wesley (1984)

[Tisseau, 2001] Tisseau, J.: Réalité virtuelle, autonomie in virtuo, Habilitation á diriger des recherches. PhD thesis (2001)

[Tolman, 1948] Tolman, E.C.: Cognitive maps in rats and men. The Psychological Review 55(4), 189–208 (1948)

[Truck, 2002] Truck, I.: Approche symbolique et floue des modificateurs linguistiques et leur lien avec l'aggregation. PhD thesis, Université de Reims, France (2002)

[Turban, 1993] Turban, E.: Decision support and expert system (1993)

[Vasan, 2003] Vasan, P.: Application of fuzzy linear "programming in production plannaning". Fuzzy Optimization and Decision Making 3, 229–241 (2003)

[Zhang, 1996] Zhang, W.: Npn fuzzy sets and npn qualitative-algebra: A computational framework for bi-cognitive modeling and multiagent decision analysis. IEEE Transactions on Systems, Man, and Cybernetics 26(8), 561–575 (1996)

[Zhang et al., 1992] Zhang, W., Chen, S., Wang, W., King, R.S.: A cognitive-map-based approach to the coordination of distributed cooperative agents, vol. 22(1), pp. 103–114 (1992)

[Zhou et al., 2003] Zhou, S., Zhang, J.Y., Liu, Z.-Q.: Quotient fcms – a decomposition theory for fuzzy cognitive maps. IEEE Transactions on Fuzzy Systems 11(5), 593–604 (2003)

Author Index