Department of Computer Science
Series of Publications A
Report A-2013-11

Methods for Redescription Mining

Esther Galbrun

To be presented, with the permission of the Faculty of Science of the University of Helsinki, for public criticism in Auditorium XV, University Main Building, on 4 December 2013, at twelve o'clock noon.

University of Helsinki
Finland

Supervisors

Professor Hannu Toivonen, University of Helsinki, Finland
Assistant Professor Mikko Koivisto, University of Helsinki, Finland

Pre-examiners

Professor Bruno Crémilleux, University of Caen, France
Professor Naren Ramakrishnan, Virginia Tech, U.S.A.

Opponent

Professor Nada Lavrač, Jožef Stefan Institute, Slovenia

Custos

Professor Hannu Toivonen, University of Helsinki, Finland

Contact information

Department of Computer Science
P.O. Box 68 (Gustaf Hällströmin katu 2b)
FI-00014 University of Helsinki
Finland

Email address: postmaster@cs.helsinki.fi
URL: http://www.cs.helsinki.fi/
Telephone: +358 9 1911, telefax: +358 9 191 51120

ISSN 1238-8645
ISBN 978-952-10-9430-9 (paperback)
ISBN 978-952-10-9431-6 (PDF)
Computing Reviews (1998) Classification: G.2.2, G.2.3, H.2.8, I.2.6
Helsinki 2013
Unigraphia

Methods for Redescription Mining

Esther Galbrun

Department of Computer Science
P.O. Box 68, FI-00014 University of Helsinki, Finland
esther.galbrun@cs.helsinki.fi
http://www.cs.helsinki.fi/people/galbrun/

PhD Thesis, Series of Publications A, Report A-2013-11
Helsinki, November 2013, 72+77 pages
ISSN 1238-8645
ISBN 978-952-10-9430-9 (paperback)
ISBN 978-952-10-9431-6 (PDF)

Abstract

In scientific investigations data oftentimes have different nature. For instance, they might originate from distinct sources or be cast over separate terminologies. In order to gain insight into the phenomenon of interest, a natural task is to identify the correspondences that exist between these different aspects.

This is the motivating idea of *redescription mining*, the data analysis task studied in this thesis. Redescription mining aims to find distinct common characterizations of the same objects and, vice versa, to identify sets of objects that admit multiple shared descriptions.

A practical example in biology consists in finding geographical areas that admit two characterizations, one in terms of their climatic profile and one in terms of the occupying species. Discovering such redescriptions can contribute to better our understanding of the influence of climate over species distribution. Besides biology, applications of redescription mining can be envisaged in medicine or sociology, among other fields.

Previously, redescription mining was restricted to propositional queries over Boolean attributes. However, many conditions, like aforementioned climate, cannot be expressed naturally in this limited formalism. In this thesis, we consider more general query languages and propose algorithms

to find the corresponding redescriptions, making the task relevant to a broader range of domains and problems.

Specifically, we start by extending redescription mining to non-Boolean attributes. In other words, we propose an algorithm to handle nominal and real-valued attributes natively. We then extend redescription mining to the relational setting, where the aim is to find corresponding connection patterns that relate almost the same object tuples in a network.

We also study approaches for selecting high quality redescriptions to be output by the mining process. The first approach relies on an interface for mining and visualizing redescriptions interactively and allows the analyst to tailor the selection of results to meet his needs. The second approach, rooted in information theory, is a compression-based method for mining small sets of associations from two-view datasets.

In summary, we take redescription mining outside the Boolean world and show its potential as a powerful exploratory method relevant in a broad range of domains.

Computing Reviews (1998) Categories and Subject Descriptors:
G.2.2 Discrete Mathematics; Graph Theory; Graph Algorithms
G.2.3 Discrete Mathematics; Applications
H.2.8 Information Systems; Database Management; Database Applications; Data mining
I.2.6 Artificial Intelligence; Problem Solving, Control Methods, and Search; Heuristic Methods

General Terms:
Algorithms, Experimentation

Additional Key Words and Phrases:
Redescription Mining, Numerical Data, Relational Query Mining, Interactive Data Mining, Pattern Set Mining

Acknowledgements

The research that led to this dissertation was conducted from 2009 to 2013 at the Department of Computer Science, University of Helsinki and the Helsinki Institute for Information Technology. I am very appreciative of the high quality and the flexibility of the working environment offered by these organizations and their personnel.

My sincere gratitude goes to Prof. Hannu Toivonen and Prof. Mikko Koivisto for taking over the role of supervisors. Also, their comments contributed to significantly improve this manuscript.

I wish to acknowledge the pre-examiners for their insightful reviews.

Along the meandering path of my doctoral studies, I have been fortunate to visit different institutions and to work with a number of skilled scholars, not least among them my coauthors on the articles listed in this dissertation. As an apprentice, I have been provided with a prime learning experience. For this I am truly thankful.

In particular, I am indebted to Dr Pauli Miettinen, for I would not have come far on this challenging journey without his critical advice and unfailing support.

Luckily, there were the casual discussions, sporadic walks and recurrent sport practices to unwind. Many thanks to the partakers who made these occasions specially enjoyable.

My mother, father and sister have been my anchor through the years.

Merci.

Original Publications of the Thesis

This thesis is based on the following peer-reviewed publications, which are referred to as Articles I–IV in the text. They are reproduced at the end of the thesis. The articles do not constitute the basis of any other doctoral dissertation. The author's contributions are described in Section 1.1.

I. Esther Galbrun and Pauli Miettinen
From Black and White to Full Color:
Extending Redescription Mining Outside the Boolean World
In *Statistical Analysis and Data Mining*, 5(4):284–303, 2012.
DOI: `http://dx.doi.org/10.1002/sam.11145`

II. Esther Galbrun and Pauli Miettinen
A Case of Visual and Interactive Data Analysis:
Geospatial Redescription Mining[1]
In *Instant Interactive Data Mining Workshop*
at the *2012 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECML/PKDD'12*
(Bristol, UK), 2012.

III. Esther Galbrun and Angelika Kimmig
Finding Relational Redescriptions
In *Machine Learning*, Published Online First, 2013.
DOI: `http://dx.doi.org/10.1007/s10994-013-5402-3`

IV. Matthijs van Leeuwen and Esther Galbrun
Compression-based Association Discovery in Two-View Data
Submitted for review.

[1]Extended version of *Siren: An Interactive Tool for Mining and Visualizing Geospatial Redescriptions*, In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'12* (Beijing,China), pages 1544–1547. ACM, 2012.

Contents

Chapter 1

Introduction

The present thesis is concerned with *redescription mining*. Roughly speaking, this data analysis task aims to find different ways of characterizing the same things and, vice versa, to find things that admit the same alternative characterizations.

As a practical example, consider the European regions of Scandinavia and Baltia. They share similar temperature and precipitation conditions and are both inhabited by the European Elk. Hence, this set of geographical areas admits two characterizations, one in terms of their climatic profile and one in terms of the occupying species.

The aim of data analysis in general is to gain useful knowledge from data, that is, to turn large amounts of data into actionable information. It is widely recognized that our understanding of a concept can be improved by considering it from different vantage points. To be more prosaic, several experiments might be carried out to study a phenomenon or, more generally, data might be available from different sources, cast in various terminologies or possess various semantics. This results in a group of datasets characterizing the same objects, known as a multi-view dataset. Then, it is of natural interest to relate and exploit these different aspects so as to better understand the concepts or phenomena at hand. This is the idea behind redescription mining.

Continuing with the example above, the data describes two different aspects of geospatial regions of Europe: their climate and their fauna. Characterizing the areas inhabited by a (set of) species in terms of the climate encountered, and the other way around, provides valuable information about the effects of climate on the species distribution. Finding such characterizations is actually an important problem in biology, known as bioclimatic niche finding [SN09, Gri17]. In this case, by providing an automated alternative to the tedious process of manually selecting species

and fitting a climatic model, redescription mining allows to explore many more combinations of conditions. Applications of redescription mining can be envisaged in a broad range of domains, including for instance social sciences and medicine.

This thesis consists of four original publications (Articles I–IV) and this introductory part. The purpose of the introduction is not to repeat the original publications. Rather, it aims to place the articles in their common context, articulate the issues addressed, and highlight the underlying transverse principles. In particular, the reader is referred to the original publications for careful review of related work, details regarding the algorithms and thorough experimental evaluation.

After providing an outline of the contributions of the original publications in the next section, we introduce the problem of mining redescriptions more thoroughly in Chapter 2. Then, in Chapters 3–5, we focus on three facets of the problem, respectively query languages, exploration strategies and pattern selection techniques. Based on these, we sketch the algorithms that constitute the main contributions of this thesis, as an opening to Chapter 6. We then proceed to illustrate the task with examples from different fields. We present results obtained with our proposed algorithms on various datasets, to illustrate the use of redescription mining in different domains and to serve as a basis for a critical discussion of the approach, before reaching conclusions in Chapter 7.

1.1 Outline of the Contributions

The contributions in this thesis are presented in the original Articles I–IV and can be summarized as follows.

I. Previously, redescription mining could handle only Boolean data, making discretization a prerequisite to using the existing tools. We extended redescription mining to categorical and real-valued attributes with a greedy algorithm that determines on-the-fly the category or interval yielding the best accuracy.

II. Building on the algorithm presented in Article I, we developed an interface for mining redescriptions from geospatial data, called Siren. We discussed desirable features of visual and interactive data analysis tools, focusing on the case of geospatial redescription mining, exemplified by Siren.

III. We introduced relational redescription mining, lifting the problem to the first order level and thereby making the approach relevant

to a new range of applications. We mine for structurally different connection patterns that describe the same object pairs in a relational dataset.

IV. Combining redescription mining with techniques from association rule mining and compression-based model selection, we present a novel method to find a small set of associations that explains how the two sides of Boolean two-view datasets are related.

The contribution of the author to all of the original publications was substantial.

Initially, Dr Pauli Miettinen suggested adapting his Greedy redescription mining algorithm to handle real-valued variables. I implemented the algorithm, largely rewriting and extending the existing code, and performed the experiments. We participated equally in writing Article I.

Later, after repeated prompting from Dr Miettinen, I set out to implement a graphical user interface for our algorithm, the matter of Article II, which we co-wrote.

Of Article III, I am the main contributor. Dr Angelika Kimmig carried out the comparison experiments with the baseline tool and edited the paper, in particular, acting as an interpreter from the field of inductive logic programming.

The collaboration on Article IV was suggested by Dr Matthijs van Leeuwen, to combine ideas from redescription mining and from his field of expertise, exceptional model mining. The problem formalization, algorithm design and article writing was done jointly. My role was very minor in implementing the algorithm. I was responsible for running most experiments.

Chapter 2

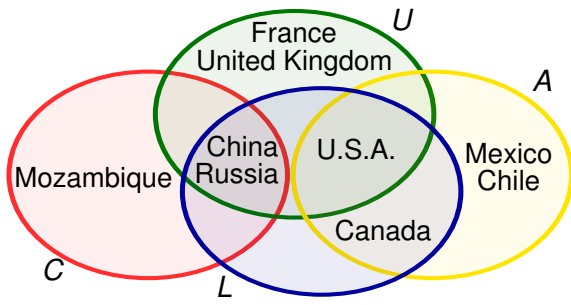
Preliminaries

In the field of data analysis, especially when considering vast amounts of data, in a process commonly called knowledge discovery in databases, *mining* usually refers to the task of extracting regularities, or patterns [HK00]. Specifically, novel, useful and understandable patterns are sought after [HMS01]. Faced with a potentially large set of factual data, obtained directly or derived from observations, for instance as the result of scientific experiments involving sensor measurements or censuses, the hope is that the analyst will be able to grasp the underlying reality by identifying recurrent patterns.

In particular, the purpose of redescription mining is to find alternative characterizations of almost the same objects. Such an approach allows to shed light on the concepts present in the dataset by identifying coherent sets of objects and related properties.

An instance of this task in the medical field could be, for example, to relate patients' background to their symptoms and to their diagnosis, so as to improve the understanding of illnesses. Revealing patterns that connect temperature and precipitation statistics to the fauna distribution constitutes another instance of the redescription mining task. Such discovery can contribute to our appreciation of the impact of climatic constraints on the habitat of these species. As mentioned previously, this pertains to the problem of ecological niche finding [SN09, Gri17].

In the relational setting, more specifically, redescription mining aims to identify correspondences between complex connection patterns, beyond the mapping of individual properties commonly considered in ontology matching and schema alignment [SE05, SAS11]. It is potentially useful for the exploration and maintenance of the massive amounts of structured information stored in knowledge bases [ABK$^+$07, CBK$^+$10, SKW07, RLT$^+$12].



A	Located in the Americas	C	History of state communism
L	Land area above 8 Billion km^2	U	Permanent member of the UN security council

Figure 2.1: Example dataset. Geographic and geopolitical characteristics of countries represented as a Venn diagram. Adapted from [PR05].

Example 1. *Throughout this introduction, we use a running example to illustrate our discussion, with variations and refinements according to the successive points of focus. Adapting the prototypical example that appeared in the earliest redescription mining publications [RKM+04, PR05, ZR05], we consider a set of nine countries as our objects of interest, namely Canada, Chile, China, France, Mexico, Mozambique, Russia, the United Kingdom and the United States of America.*

Following Parida and Ramakrishnan [PR05], let us start with a simple toy dataset consisting of four properties characterizing these countries, represented as a Venn diagram in Figure 2.1. Consider the pair of statements below:

- *Country outside the Americas with land area above 8 billion square kilometers.*
- *Permanent member of the UN security council with a history of state communism.*

Both statements are satisfied by the same countries, namely China and Russia. They constitute alternative characterizations of the same subset of countries in terms of geographic and geopolitical properties, respectively. Hence, they form a redescription.

We now provide a formal albeit very general definition of redescriptions and the redescription mining task.

2.1 Problem Definition

Let $\mathcal{O}$ be a set of elementary *objects* and $\mathcal{A}$ a set of *attributes* characterizing properties of the objects or of relations between them. The attributes arise

from different sources, terminologies, etc., represented as a set of *views* V. A function υ maps each attribute to the view to which it belongs, $\upsilon : \mathcal{A} \to V$. The dataset is fully specified by the triplet $(\mathcal{O}, \mathcal{A}, \upsilon)$.

A logical statement expressed over attributes in $\mathcal{A}$ and evaluated against the dataset forms a *query*. A query language Q is a set of valid queries. To evaluate a statement against a dataset is to replace the variables in the statement by objects from the dataset and identify the substitutions for which the ground formula holds true. This subset of objects or of non-empty tuples of objects is called the *support* of query q and denoted as $\text{supp}(q)$. The set of *entities*, denoted as E, consists of all the possible substitutions for queries in Q. The set of attributes appearing in query q is denoted by $\text{att}(q)$ and we overload the function υ to also denote the union of their views, $\upsilon(q) = \bigcup_{A \in \text{att(q)}} \upsilon(A)$. As a straightforward means to ensure that two queries provide different characterizations, their attribute sets are required to be disjoint. We consider a symmetric binary relation $\sim$ over the power set of entities as a Boolean indicator of support similarity. Finally, we denote by $\mathcal{C}$ a set of arbitrary constraints that can be used to specify a bias towards certain queries.

In this context, we define a redescription as follows.

Definition 1. *Given a dataset $(\mathcal{O}, \mathcal{A}, \upsilon)$, a query language Q over $\mathcal{A}$ and a binary relation $\sim$, a* ***redescription*** *is a pair of queries $(q_A, q_B) \in Q \times Q$ such that $\upsilon(q_A) \cap \upsilon(q_B) = \emptyset$ and $\text{supp}(q_A) \sim \text{supp}(q_B)$.*

And redescription mining is simply the task of finding such pairs.

Problem 1 (Redescription Mining)**.** *Given a dataset $(\mathcal{O}, \mathcal{A}, \upsilon)$, a query language Q over $\mathcal{A}$, a binary relation $\sim$, and a set $\mathcal{C}$ of constraints, find all redescriptions that satisfy constraints in $\mathcal{C}$.*

Example 2. *Let us consider the example above in light of this terminology. The objects are nine countries, and the four attributes can be divided into two views, say $\boldsymbol{G}$ and $\boldsymbol{P}$, corresponding to the domains of geography, i.e. attributes A and L, and geopolitics, i.e. attributes C and U, respectively. That is, we have that $\mathcal{A} = \{A, C, L, U\}$ and, for instance, $\upsilon(A) = \boldsymbol{G}$. The first statement forms a query over geographic attributes, which could be equivalently written as $q_{\boldsymbol{G}} = \neg A \wedge L$. When evaluated on the dataset, it is satisfied by two countries. Specifically, we have $\text{supp}(q_{\boldsymbol{G}}) = \{\textit{China}, \textit{Russia}\}$. The second query, over geopolitical attributes, $q_{\boldsymbol{P}} = U \wedge C$, has the same support. Thus, for any reasonable choice of similarity relation $\sim$ we have that $\text{supp}(q_{\boldsymbol{G}}) \sim \text{supp}(q_{\boldsymbol{P}})$. Since in addition $\upsilon(q_{\boldsymbol{G}}) \cap \upsilon(q_{\boldsymbol{P}}) = \{\boldsymbol{G}\} \cap \{\boldsymbol{P}\} = \emptyset$, then $(q_{\boldsymbol{G}}, q_{\boldsymbol{P}})$ is a redescription.*

Here, we roughly consider data analysis methods with a focus on description to belong to data mining, while those methods with a focus on prediction are categorized into machine learning. In other words, the former area consists of techniques that aim at detecting regularities in the data, i.e. patterns, emphasizing the interpretability of the results, while the latter comprises techniques that aim to predict some properties or relations of unseen objects given a subset of observed ones. However, patterns resulting from data mining can constitute the building blocks of learning systems by providing higher level features, while machine learning tasks such as classification or regression can be found at the core of pattern discovery systems. The line between the two areas is easily blurred.

As its name suggests, redescription mining is a descriptive data analysis problem, a data mining task. Because the aim is not to learn a model to predict unseen data, but rather, to describe the data at hand as well as possible, the expressivity and interpretability of the results are particularly emphasized.

In our context, expressivity and interpretability are to be understood in the following acceptations: The variety of concepts that a language can represent determines its expressive power, or expressivity, while the interpretability of an element of the language relates to the ease with which the associated meaning can be apprehended. Interpretability is difficult to measure but is promoted by favoring concise, simple queries. As a consequence of this accent put on the descriptive aspect, certain query languages are more adequate for the task than others. In particular, throughout our work we adopt the following position with respect to query languages. Our preference goes to Boolean formulae specifying explicit constraints on the range of individual attributes. Linear functions defined over the attributes, on the other hand, are deemed to have limited interpretability and hence to be unsuitable for our purpose.

In any given instance of the redescription mining task, we consider a collection $\mathcal{O}$ of elementary objects, sometimes also referred to as samples. The dataset consists of attributes in $\mathcal{A}$ characterizing the properties of these objects and possibly of the relations linking them, as well. That is, we consider both propositional and relational datasets. These two settings are formalized and discussed in more depth together with the associated query languages in Chapter 3.

The set of views V represents the various sources, domains or terminologies from which the data originate. For instance, the attributes of our toy dataset above can be naturally split between geography (A and L) and geopolitics (C and U), while additional attributes could stem from the eco-

nomic, social or cultural domains. Climatic conditions on one hand and fauna on the other hand form two clearly distinct sets of attributes in biological niche finding, our example in the field of biology, while in the field of medicine, the objects of the study might be a set of patients with their personal background information, symptoms and elements of diagnosis, as three different views, for instance.

The purpose of redescription mining is to find *alternative characterizations* of almost the same objects, so as to relate concepts across different sources, domains or terminologies. Therefore, we require that the attributes over which both queries of a redescription are expressed come from disjoint sets of views, i.e. $v(q_A) \cap v(q_B) = \emptyset$, and say that such queries are *structurally different*. If two statements are logically equivalent, then the associated queries trivially have the same support. Uninteresting tautologies of this kind are ruled out by this requirement.

Conceptually, the number of views can be arbitrarily large. Yet, two settings are generally preferred. At one extreme, each individual attribute can be seen to form a separate view. This corresponds to the case where all attributes are gathered into a single dataset and the restriction that none of them appears in both queries simultaneously. At the other extreme, we might consider the attributes to be naturally split into two subsets that each constitutes a view.

From now on, we will focus primarily on the case where the attributes are split between two views, arbitrarily denoted as $\mathbf{L}$ (for left-hand side) and $\mathbf{R}$ (for right-hand side). That is, we have two subsets of attributes,

$$\mathcal{A}_{\mathbf{L}} = \{A \in \mathcal{A}, v(A) = \mathbf{L}\} \text{ and } \mathcal{A}_{\mathbf{R}} = \{A \in \mathcal{A}, v(A) = \mathbf{R}\},$$

such that $\mathcal{A}_{\mathbf{L}} \cup \mathcal{A}_{\mathbf{R}} = \mathcal{A}$ and, trivially, $\mathcal{A}_{\mathbf{L}} \cap \mathcal{A}_{\mathbf{R}} = \emptyset$. Then, a pair of structurally different queries simply consists of queries $q_{\mathbf{L}}$ and $q_{\mathbf{R}}$ expressed over $\mathcal{A}_{\mathbf{L}}$ and $\mathcal{A}_{\mathbf{R}}$, respectively. Still, discussions on this setting, known as two-view or two-fold setting, extend naturally to other settings as well.

In the presence of multiple views, the correspondence between the elementary objects across the views might not be available. It might be that the sets of objects occurring in different views are not identical, that some objects are associated with redundant observations in one view, or that a single object from one view corresponds to multiple objects in another view, as is the case for instance of geospatial measurements with varying scales. Establishing the mapping between the views can be nontrivial and constitutes a problem of its own [TKOK11], ignored here. We assume that the input data consist of aligned views, i.e. that the bijection of objects across the views is known. Moreover, except for missing values which we

consider in Article I, dealing with measurement errors, sampling bias and similar problems affecting the data quality is outside the scope of our work.

The purpose of redescription mining is to find alternative characterizations of *almost the same objects*. Thus, the similarity of the supports of the queries is a defining quality of a redescription. We say that a pair of queries is *accurate* if their supports are similar. In general, the similarity relation $\sim$ between support sets is specified as a set similarity function f together with a threshold σ, such that $E_a \sim E_b \iff f(E_a, E_b) \geq \sigma$. Specifically, the Jaccard coefficient is a common choice for f, as we mention in Section 5.1.1.

Nevertheless, structural difference and accuracy are typically not sufficient to guarantee the interestingness of the result. Other criteria also impact its quality. The set of constraints on the queries and their supports, $\mathcal{C}$, allows to specify a bias towards certain redescriptions, in particular to include background knowledge. Furthermore, we are generally looking for a set of redescriptions, the interestingness of which we want to evaluate not just individually but as a whole. Interestingness and the selection of high quality redescriptions constitute the focus of Chapter 5.

2.2 Related Work

Redescription mining is a multi-view data mining technique in that it exploits multiple views on the objects to identify interesting patterns. Similar ideas motivate multi-view learning approaches. Pioneered by Yarowsky [Yar95] in the context of word sense disambiguation, and by Blum and Mitchell [BM98] under the term co-training, the principle of exploiting distinct views to strengthen learning algorithms has attracted increasing interest. It has been applied to clustering [NG00, BS04, BS05], support vector machines [FHM+05], canonical correlation analysis [KK08] or factor analysis [VKKK12], among others. Arguably, redescription mining can be seen as the data mining pendant of multi-view techniques in machine learning.

Other data mining problems such as emerging patterns [NLW09], subgroup discovery [UZT+09] or exceptional model mining [LFK08] can also be seen as multi-view approaches, although they are rarely presented from that perspective. Indeed, the common aim of these techniques is to find queries over one view defining a set of objects with an uneven distribution in the other view when compared to the remaining objects. That is, these approaches consider one view as description and the other as target. In this

context, the symmetricity of its approach, in other words, the fact that the views are treated equally, is a distinguishing feature of redescription mining.

The relational setting is tightly related to inductive logic programming concepts in general [DR08, MDR94, QCJ93, Mug95, Sri07] and multi-relational query mining in particular [DT99, DRR04], from which terminology and notation are borrowed. The work presented here draws on techniques from various other areas of data analysis, such as, in particular, graph mining, constraint-based mining or model selection. References are provided in the relevant chapters and the original articles.

Redescription mining was introduced by Ramakrishnan et al. [RKM+04] in 2004. They proposed to find set theoretic expressions over indicator functions that define similar subsets of elements. There, indicator functions and set theoretic expressions are used as representation, equivalently to Boolean attributes and logical statements.

Several conceptually related problems are discussed in [PR05], including *story telling* [RKM+04], the task of finding a succession of approximate redescriptions such that the first and last are supported by two distinct given sets of entities. Another kindred task, finding *straddling-biclusters* [JMR08], combines mining redescriptions and biclusters. Finally, *query by output* is a closely related problem in the area of database systems [TCP09]. It aims at finding an instance-equivalent query for a given input query, or in the words of redescription mining, completing a query pair to form an accurate redescription. Considering SQL queries, it requires to determine not only good selection predicates but also relevant relations and projections.

The approaches proposed for redescription mining have been based on various ideas, including decision trees [RKM+04, Kum07], Karnaugh maps [ZR05], co-clusters [PR05], frequent itemsets [ZR05, GMM08] and greedy search [GMM08]. However, they all focused on Boolean propositional attributes, restricting the applicability of the method. Indeed, as a consequence, discretization, binarization or propositionalization were required preliminaries in order to mine redescriptions from datasets containing non-Boolean or relational attributes. Such preprocessing procedures typically require extensive domain knowledge, entail an information loss that can impact the subsequent analysis and greatly inflate the search space.

To address this issue, we extended the Greedy algorithm [GMM08] with efficient on-the-fly discretization in its innermost loop, allowing to handle nominal and real-valued attributes, as reported in Article I. Next, we designed an algorithm to mine redescriptions from network data, presented

in Article III, using an alternating scheme similar to that presented by Ramakrishnan et al. [RKM+04].

Main strategies for mining redescriptions are discussed in Chapter 4.

To summarize, redescription mining automatically identifies both a pair of queries and an associated set of objects, neither of which needs to be specified in advance. The results of redescription mining offer a dual perspective. On one hand, the queries of a redescription contain attributes that are related to each other, as they can be used to characterize the same set of objects. On the other hand, the support of the redescription defines a particularly coherent set of objects, as it admits alternative characterizations.

So far we presented the problem of redescription mining from a conceptual point of view, deliberately keeping the discussion at a fairly high abstraction level. We will now focus on some more practical aspects, in other words, on choices that need to be made when implementing the general principle.

The three following chapters are organized around three primary facets of the problem that can be considered independently from each other, to some extent. First, we discuss query languages in Chapter 3, where possible choices for Q are defined formally. Second, we sketch strategies to explore the space of query pairs in search for redescriptions, in Chapter 4. Third, in Chapter 5, we look at what constraints $\mathcal{C}$ can be used to characterize good redescriptions and study pattern selection techniques more generally.

Chapter 3

Query Languages

In this chapter, we explore variations on a theme and are concerned with choices of query languages. More precisely, we define the queries that are used in redescription mining, offering means to represent logical combinations of constraints on the range of individual attributes.

The formalism presented here corresponds to that previously used in Boolean propositional redescription mining [PR05, GMM08], which we adapted to support other types of attributes and later extended to the relational case, respectively in Article I and Article III.

Queries consist of logical statements evaluated against the dataset. The statements are obtained by combining atomic predicates built from individual attributes using Boolean operators. Replacing predicate variables by objects from the dataset and verifying whether the conditions of the predicates are satisfied returns a truth value. The objects or object tuples in substitutions satisfying the statement constitute the support of the query.

A query language is a set of acceptable queries, dependent on the supported types of attributes, the principles for building predicates and the syntactic rules for combining them into statements. We discuss, in turn, propositional and relational queries.

3.1 Propositional Queries

A *propositional dataset* consists of attributes characterizing properties of individual objects. We generally consider a homogeneous set of objects, in the sense that each attribute applies to all objects, regardless of possible missing values. In that context, the values taken by the attributes in $\mathcal{A}$ for each of the objects are collected into a matrix D with $|\mathcal{O}|$ rows, one per

Table 3.1: Example dataset. Geographic attributes of world countries: localization in the southern hemisphere (1), existence of oceanic borders (2–4), continental location (5), land area (6) and maximum elevation (7).

Country	1) South Hemisphere	2) Atlantic Ocean	3) Indian Ocean	4) Pacific Ocean	5) Continent	6) Area (10^9 km^2)	7) Elev. (m)
CA	false	true	false	true	{North America}	9.98	5959
CL	true	true	false	true	{South America}	0.76	6893
CN	false	false	false	true	{Asia}	9.71	8850
FR	false	true	false	false	{Europe}	0.64	4810
GB	false	true	false	false	{Europe}	0.24	1343
MX	false	true	false	true	{North America}	1.96	5636
MZ	true	false	true	false	{Africa}	0.79	2436
RU	false	false	false	true	{Asia, Europe}	17.10	5642
US	false	true	false	true	{North America}	9.63	6194

object, and $|\mathcal{A}|$ columns, one per attribute. In other words, $D(i, j) = A_j(o_i)$ is the value of attribute $A_j \in \mathcal{A}$ for object $o_i \in \mathcal{O}$.

As a special case, we say that the dataset is geospatial when the objects are associated to spatial coordinates, that is, when they can be located in a spatial reference system. Then, the support of the resulting geospatial queries and geospatial redescriptions over that dataset can be naturally represented on a map.

Example 3. *Refining Example 2, consider the dataset shown in Table 3.1. Again, the set of objects is a subset of world countries. Each of the seven attributes corresponds to a geographic property: localization in the southern hemisphere (1), existence of a border to the Atlantic Ocean (2), to the Indian Ocean (3), or to the Pacific Ocean (4), continental location (5), land area in billion squared kilometers (6) and maximum elevation in meters (7), referring to the mainland area only. This data can be represented as matrix G with ten rows and seven columns. Furthermore, we can identify attributes with the columns of this matrix. For instance, the first attribute, localization in the southern hemisphere, is denoted as the corresponding vector G_1.*

This constitutes a simple example of a geospatial dataset, since the objects, i.e. the countries, can be associated to spatial coordinates such as latitude and longitude (omitted here).

Predicates are constructed over individual attributes and combined into statements to form the queries.

3.1.1 Predicates

The values that an attribute can take constitute its range. A predicate is constructed from an attribute by restricting the values to a selected subset of its range. Consider an attribute $A_j \in \mathcal{A}$ with range R. By fixing a subset $R_S \subseteq R$ we can turn the associated data column into a truth value assignment, that is, into a Boolean vector indicating which values are within the specified range. Using Iverson bracket, this is denoted as $[A_j \in R_S]$. In effect, this selects the subset of objects for which attribute A_j takes value in R_S, $s(A_j, R_S) = \{o_i \in \mathcal{O}, A_j(o_i) \in R_S\}$ and $[A_j \in R_S]$ is an indicator of membership in this subset.

Depending on their range, object attributes can be classified into types. In turn, we consider three types of attributes, Boolean, nominal and real-valued and the predicates constructed from them.

Boolean predicates. Boolean attributes take value either `false` or `true`. Equivalently, their range is $\{0, 1\}$. A Boolean attribute can be interpreted as a truth value assignment for the objects in a natural way and thus directly yields a predicate. For Boolean attributes we omit the bracket notation and simply denote the Boolean predicate $[A = \texttt{true}]$ by A. Typically, we do not consider the complementary assignment $[A = \texttt{false}]$ as it can be equivalently obtained with negation.

In our example, G_4 is a Boolean attribute that corresponds to a predicate with the following truth assignment on this dataset:

$$\langle 1, 1, 1, 0, 0, 1, 0, 1, 1 \rangle ,$$

selecting the six countries bordering the Pacific Ocean.

Nominal predicates. An attribute A whose range is an unordered set C, or its powerset, is called a nominal (or categorical) attribute, single-valued or multi-valued respectively. The elements of C are called the categories of attribute A. A truth value assignment is obtained by choosing a subset of the categories $C_S \subseteq C$ or a single category $c \in C$, to select objects that belong to these categories. The corresponding nominal predicates are denoted as $[A \in C_S]$ and $[A = c]$, respectively.

The first attribute from our example, the continental location, is a nominal attribute. Four countries located in the Americas satisfy the predicate

$$[G_5 \in \{\text{North America}, \text{South America}\}] ,$$

corresponding to the truth assignment $\langle 1, 1, 0, 0, 0, 1, 0, 0, 1\rangle$. In this case, the attribute is multi-valued because objects can be associated to multiple categories: Russia spans over both Europe and Asia.

Practically, only single-valued nominal attributes and predicates with an individual category are considered. Multi-valued nominal attributes are equivalently represented with several Boolean attributes, one per category.

Real-valued predicates. An attribute A whose range is a subset of the real numbers $R \subseteq \mathbb{R}$ is a real-valued attribute. A truth value assignment for the objects can be obtained by choosing any subset of R. However, for interpretability reasons, it is typically constructed based on a single contiguous subset of R, i.e. an interval $[a, b] \subseteq R$, and denoted as $[a \leq A \leq b]$.

Notice that for a given real-valued attribute, there can be an infinity of intervals yielding the same truth-value assignment. Consider for example the sixth attribute, the land area, corresponding to the following vector

$$G_6 = \langle\ 9.98,\ 0.76,\ 9.71,\ 0.64,\ 0.24,\ 1.96,\ 0.79, 17.10,\ 9.63\rangle\ .$$

Any pair (λ, ρ) where $\lambda \in]0.79, 1.96[$ and $\rho \in]9.98, 17.10[$ will yield the same truth value assignment

$$[\lambda \leq G_6 \leq \rho] = \langle\quad 1,\quad 0,\quad 1,\quad 0,\quad 0,\quad 1,\quad 0,\quad 0,\quad 1\rangle\ .$$

Hence, the definition of a consistent query language must include a criterion for selecting one among these equivalent intervals. Yet, it is disputable whether $[1 \leq G_6 \leq 10]$, because it has rounded bounds, is a better choice than tighter $[1.96 \leq G_6 \leq 9.98]$, and similarly, whether $[G_6 \leq 10]$ should be preferred over equivalent $[0.24 \leq G_6 \leq 10]$, for instance. Indeed, in both cases the two intervals correspond to the same truth assignment and favoring one over the other depends, in particular, on whether rounded bounds are considered more interpretable than tight intervals, or vice versa.

3.1.2 Statements

Predicates make up the building blocks of statements. Propositional predicates can be combined using Boolean operators, negation ($\neg$), conjunction ($\wedge$) and disjunction ($\vee$). The truth assignment for the associated query is obtained by combining the truth assignment of the individual predicates accordingly. Equivalently, the individual truth assignments define subsets of objects that can be combined by means of the corresponding set operators, complement ($\setminus$), intersection ($\cap$) and union ($\cup$). The resulting subset of objects is the support of the query.

For instance, in the context of Example 3, the query

$$q_1 \;=\; G_4 \wedge \neg\, G_1 \wedge [6000 \leq G_7]$$

describes countries bordering the Pacific outside the southern hemisphere where the highest peak reaches over 6000 meters. Two countries, China and the U.S.A., satisfy these conditions, that is, support this query.

In the propositional case, the possible substitutions for the queries, called the entities, are simply individual objects, $E = \mathcal{O}$.

Monotone conjunctions. Monotone conjunctions are the most restricted query language, where predicates are combined using only conjunction operators. For example, the following query is a monotone conjunction

$$q_2 \;=\; G_3 \wedge G_2 \wedge [1000 \leq G_7 \leq 2000]\ ,$$

while q_1 above does not belong to this language since it is not monotone.

Conjunctive monotone queries directly correspond to itemsets where each predicate is an item. Itemsets have been extensively studied in the literature, especially to design efficient frequency based mining algorithms [HCXY07, Goe03]. In particular, they are easily arranged in a partial order based on inclusion and verify the downward closedness property. That is, if query q_i is a subset of query q_j, then the support of q_i is a superset of the support of q_j. As a consequence, the search space of this query language can be explored efficiently in a level-wise fashion.

The query language used in Article IV consists of monotone conjunctive queries over Boolean predicates, which, being the most restricted also affords efficient exhaustive search.

Monotone conjunctions are at the lower end of the scope of propositional queries, at the same time easy to interpret and to find. But excluding negations and disjunctions severely limits the expressivity.

Unrestricted queries. At the other end of the scope are unrestricted propositional queries, in which predicates can be combined using any of the three operators with no other limits than the usual rules of algebra. For example

$$\begin{aligned}
q_3 \;&=\; (G_3 \vee G_2) \wedge \neg\, G_1\ ,\\
q_4 \;&=\; (\neg\, G_2 \wedge [G_5 = \text{Asia}]) \vee ([5000 \leq G_7] \wedge G_4)\ ,\\
q_5 \;&=\; (G_1 \wedge \neg\, [4000 \leq G_7 \leq 6000]) \vee [3000 \leq G_7]\ \text{, and}\\
q_6 \;&=\; \big(\neg\, \big(G_3 \vee ([G_5 = \text{Europe}] \wedge G_2)\big) \wedge [0.3 \leq G_6 \leq 0.9]\\
&\quad \wedge G_1\big) \vee \big([G_7 \leq 6300] \wedge G_4\big)\ ,
\end{aligned}$$

```
 <query>   →  (<query>) ∧ <literal>
 <query>   →  (<query>) ∨ <literal>
 <query>   →  <literal>
<literal>  →  <predicate>
<literal>  →  ¬ <predicate>
```

Figure 3.1: Generative grammar of the linearly parsable propositional query language. The non terminal symbol `<predicate>` is a predicate as defined in Section 3.1.1.

as well as q_1 and q_2 above, all belong to this query language.

However, while permitting full expressivity of Boolean formulae, this unrestricted query language contains queries that are difficult to interpret, for instance because of deeply nested structures.

Consider, as a simple example, a query over numerous attributes, possibly with a complex nested structure, such as q_6. Its support might match very well that of another query, resulting in a highly accurate redescription. However, because of the many entangled conditions, it will be difficult for the analyst to interpret it, that is, to understand the conveyed meaning, directly limiting its interestingness.

In addition, the resulting space of redescriptions lacks organizing structure and is therefore very hard to search.

Linearly parsable queries. As a compromise between these two extremes, we propose in Article I to use linearly parsable formulae as our propositional query language. This language comprises the queries generated by the simple formal grammar shown in Figure 3.1, where the non terminal symbol `<predicate>` is a predicate as defined in Section 3.1.1.

Simply put, these are queries which can be evaluated from left to right irrelevant of the binary operators precedence. Among the example queries q_1–q_6, all but q_4 and q_6 satisfy this criterion.

As an additional requirement of the language to ensure better interpretability, we restrict every attribute to appear only once. Query q_5 will be rejected since attribute G_7 appears twice. However, in this case, q_5 can be equivalently rewritten in the acceptable form $G_1 \vee [3000 \leq G_7]$.

Although in theory the choice of a query language is a building block of the problem definition, prior to the algorithm design, computability represents a strong practical constraint influencing the choice. For instance, linearly parsable queries naturally result from iterative atomic extensions, progressively appending literals, i.e. positive or negated predicates, to the

current query, as it happens in the Greedy algorithm [GMM08]. Another example are the queries obtained with the CARTwheels algorithm [RKM+04], whose typical form directly reflects the decision trees used for mining them.

3.2 Relational Queries

When the dataset contains information about the relations between objects in addition to, or instead of, the properties of individual objects, it is called a *relational dataset*. An interaction between n objects is modelled as an n-ary relation, corresponding to a hyperedge of cardinality n in a hypergraph whose nodes represent the objects.

In the work presented here, we restrict ourselves to binary relations. That is, we consider only interactions involving two objects at once, as can be represented by usual graph edges. This restriction allow us to employ techniques from graph mining when processing the datasets. Indeed, this kind of relational dataset can be viewed as a multilabelled directed graph $(\mathcal{O}, \mathcal{R})$, where nodes correspond to the objects $\mathcal{O}$, and edges to relations $\mathcal{R}$ between them. Two families of functions, $\mathcal{N}$ and $\mathcal{E}$, label nodes and edges with their attributes, respectively. Relations of higher arity might be decomposed into binary relations, possibly by introducing intermediary objects.

Similarly to the propositional setting, predicates can be constructed from the attributes and combined into statements to form relational queries. We introduced relational queries for redescription mining in Article III.

Example 4. *Continuing with our example on world countries, we now look at a dataset representing their relations from a geopolitical point of view. This dataset involves other objects in addition to the nine countries: five international organizations, namely the Commonwealth of Nations, the European Union (EU), the North Atlantic Treaty Organization (NATO), the Organization of American States (OAS) and the United Nation Security Council, as well as five cities and seven languages.*

The relations existing between these objects can be represented as a directed and labelled network, as shown in Figure 3.2. Object attributes could be indicated as labels on the nodes. However, they are listed separately in Table 3.2 for better readability.

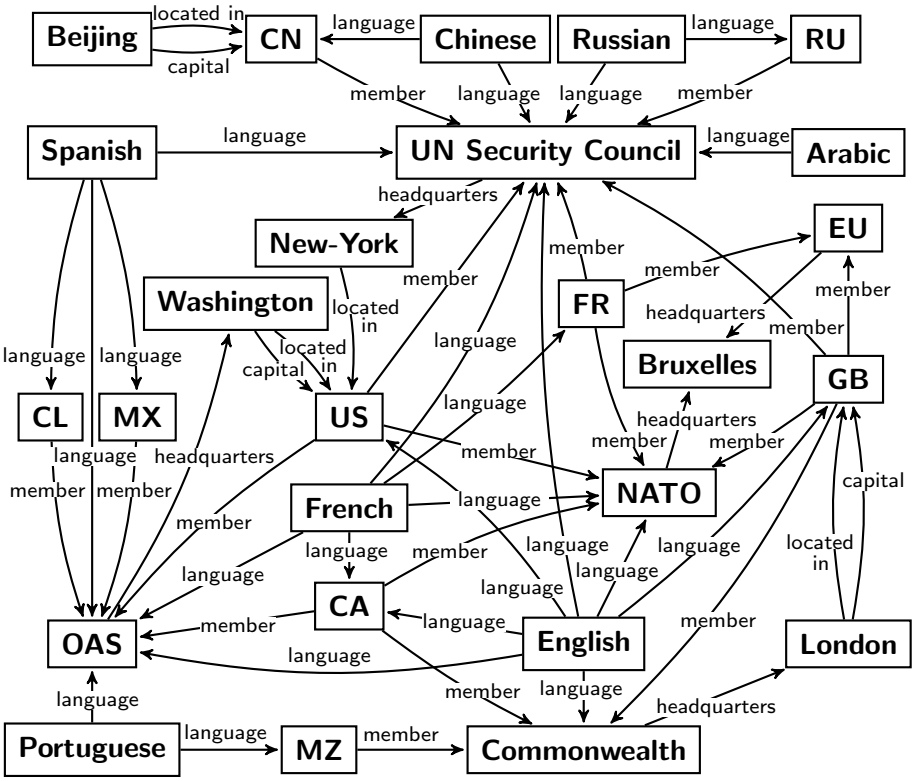


Figure 3.2: Example dataset. Geopolitical relations between world countries involving cities, international organizations and languages.

3.2.1 Predicates

A relational dataset might be heterogeneous, in the sense that not all attributes are defined for every object. The subset of objects that an attribute characterizes constitute its domain, such that $\mathrm{dom}(N_i) \subseteq \mathcal{O}$ for node attributes and $\mathrm{dom}(E_i) \subseteq \mathcal{O} \times \mathcal{O}$ for edge attributes.

For instance, in our example, population is recorded for both countries and cities and the year of foundation is defined for organizations only. All attributes are not gathered into a single matrix. Here, they are presented in distinct tables for the countries, cities and international organizations, in Tables 3.2 (a), (b) and (c), respectively.

Similarly to the propositional predicates seen previously, node and edge predicates can be built from object and relation attributes, respectively. In addition, we also consider comparison predicates.

Table 3.2: Example dataset. Geopolitical attributes.

(a) World countries.

Country	1) History of Communism	2) History of Colonialism	3) Political Regime	4) Population (10^6 hab.)
CA	`false`	`false`	Monarchy	33.476
CL	`false`	`false`	Republic	16.572
CN	`true`	`false`	Republic	1 353.821
FR	`false`	`true`	Republic	65.350
GB	`false`	`true`	Monarchy	63.181
MX	`false`	`false`	Republic	115.296
MZ	`true`	`false`	Republic	22.894
RU	`true`	`false`	Republic	143.300
US	`false`	`false`	Republic	315.550

(b) Cities.

City	6) Population (10^6 hab.)
Beijing	16.801
Bruxelles	1.119
London	8.173
New-York	8.336
Washington D.C.	5.703

(c) International organizations.

Organisation	7) Year of Foundation
Commonwealth	1926
EU	1952
NATO	1949
OAS	1948
UN Security Council	1946

Node predicates. For a given node attribute N_i and subset R_S of its range, a node predicate $\nu_{N_i}^{R_S}(o)$ is true for an object o if and only if the node attribute N_i is defined and takes value in R_S for this object. Node predicates are the counterpart of propositional predicates, with the additional condition that the attribute needs to be defined, which is implicitly assumed in the propositional case. Using Iverson bracket notation, this is written as

$$\nu_{N_i}^{R_S}(o) = [o \in \mathrm{dom}(N_i) \ \wedge \ N_i(o) \in R_S] \ .$$

For example, objects for which population information is available and ranges from 10 to 30 millions support the predicate $\nu_{\texttt{population}}^{[10,30]}(o)$, that is, Chile, Mozambique and Beijing. Node predicates $\nu_{\texttt{independence}}^{[1800,1900]}(o)$ and $\nu_{\texttt{regime}}^{Monarchy}(o)$ respectively select countries that became independent during the XIX$^{\text{th}}$ century and monarchies, namely Canada, Chile and Mexico, on one hand, Canada and the United Kingdom on the other. In the latter case, we slightly abuse notation to denote, strictly speaking, the singleton set $\{Monarchy\}$.

Edge predicates. Likewise, for a given edge attribute E_i and subset R'_T of its range, an edge predicate $\epsilon_{E_i}^{R'_T}(o_1, o_2)$ is true for a pair of objects (o_1, o_2) if and only if the edge label E_i is defined for that pair and takes value in R'_T. This is equivalently expressed as

$$\epsilon_{E_i}^{R'_T}(o_1, o_2) = [(o_1, o_2) \in \mathrm{dom}(E_i) \ \wedge \ E_i(o_1, o_2) \in R'_T] \ .$$

When the range of the attribute is limited to a single value it simply indicates the existence of the relation without qualifying it. In our example, all edge attributes are of such existential type. In that case, we denote the corresponding predicate simply as $\epsilon_{E_i}(o_1, o_2)$.

In particular, the edge predicate $\epsilon_{\texttt{language}}(o_1, o_2)$ selects pairs of objects where the former is an official language of the latter. There are 21 such pairs in our example dataset, including the pairs of languages and organizations (Arabic, UN Security Council), (Spanish, OAS) and (French, NATO), as well as the pairs of languages and countries (Spanish, Mexico) and (Chinese, China). The membership predicate $\epsilon_{\texttt{member}}(o_1, o_2)$ is supported by 18 pairs of objects, a country and an organization, where the country is a member of the organization.

In comparison to these existential edge attributes we could consider a detailed variant, e.g. an attribute which does not simply indicate membership but more precisely qualifies the relation with values such as permanent, observing, or elected.

Comparison predicates. Finally, comparison predicates are built as follows. For a given object attribute we choose as a comparison function a binary relation $\prec$ defined over its range. Then, a comparison predicate $\phi_{N_i}^{\prec}(o_1, o_2)$ is true for a pair of objects (o_1, o_2) if and only if both node labels $N_i(o_1)$ and $N_i(o_2)$ are defined and $N_i(o_1) \prec N_i(o_2)$. That is, a comparison predicate is defined as

$$\phi_{N_i}^{\prec}(o_1, o_2) = [o_1 \in \mathrm{dom}(N_i) \ \wedge \ o_2 \in \mathrm{dom}(N_i) \ \wedge \ N_i(o_1) \prec N_i(o_2)] \ .$$

As an example, consider the less-than relation over the real-valued population attribute, such that the comparison predicate $\phi_{\texttt{population}}^{<}(o_1, o_2)$ holds true if and only if o_1 is less populated than o_2. The pairs (Chile, Beijing), (London, New-York) and (Canada, Russia), among others, support this predicate.

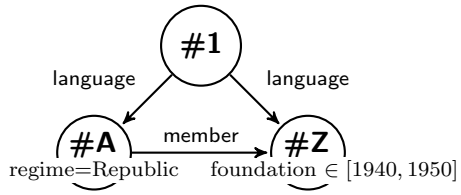


Figure 3.3: Graphical representation of a relational query $q_1(\#A, \#Z)$.

3.2.2 Statements

As in the propositional case, the predicates introduced in the previous section constitute building blocks, which are combined into statements to form queries.

In the relational setting, queries consist of monotone conjunctions of nodes, edges and comparison predicates with a subset of variables of interest selected as query variables. More precisely, borrowing terminology from inductive logic programming [MDR94] and using the *Prolog* notation [DEDC96], a relational query is a definite clause of the form

$$q(X_1, \ldots, X_m) \text{ :- } b_1, \ldots, b_n \,.$$

where the body elements b_i are node, edge or comparison predicates and q is a special predicate denoting the query. The query variables $X_1, \ldots, X_m$ in the head also occur in the body.

Such queries can be represented in graphical form. For this purpose, we adopt the following conventions. While data nodes are represented as squares, variables, i.e. query nodes, are represented as circles. Furthermore, we use the hash symbol together with a letter to denote the query variables and with a number to denote any other intermediate variable.

For instance, the graph in Figure 3.3 represents the following relational query

$$\begin{aligned} q_1(\#A, \#Z) \text{ :- } & \nu^{Republic}_{\texttt{regime}}(\#A), \nu^{[1940,1950]}_{\texttt{foundation}}(\#Z), \epsilon_{\texttt{member}}(\#A, \#Z), \\ & \epsilon_{\texttt{language}}(\#1, \#A), \epsilon_{\texttt{language}}(\#1, \#Z) \,. \end{aligned}$$

This query involves node predicates based on nominal and numerical attributes. However, we generally consider only Boolean and nominal object attributes to construct node predicates, while numerical attributes are used to construct comparison predicates.

To determine the support of the query, the statement is matched against the data: each variable in the query has to be matched to a node in the graph, respecting the predicates in the query body. We denote such

a match of variables Y_j to objects o_{i_j} by the corresponding substitution $\theta = \{Y_1/o_{i_1}, \ldots, Y_l/o_{i_l}\}$; θ reduced to the query variables is called *answer substitution*. The set of all distinct answer substitutions of query q is its support, $\mathrm{supp}(q)$. Hence, the support of query $q(X_1, \ldots, X_m)$ is a set of m-tuples of objects.

The support of query q_1 above consists of pairs of objects, a republic and an organization founded in the 1940s, where the former is a member of the latter and they share an official language. Such is the case of Chile and the OAS or Canada and the NATO, among others.

As another example, by adding the intermediate variable #1 to the head as a query variable renamed as, say, $\#B$, we obtain a query of arity three, $q_2(\#A, \#B, \#Z)$. Entities supporting this modified query are triplets consisting of a country, a language and an organization, including (Chile, Spanish, OAS) or (Canada, English, NATO) as well as (Canada, French, NATO).

Further examples are shown in Figure 3.4 respectively representing the following relational queries:

$$
\begin{aligned}
q_3(\#A) \;&:\!-\; \epsilon_{\texttt{capital}}(\#A, \#1), \epsilon_{\texttt{headquarters}}(\#2, \#A)\,. \\
q_4(\#A, \#Z) \;&:\!-\; \epsilon_{\texttt{capital}}(\#1, \#A), \epsilon_{\texttt{located in}}(\#2, \#A), \\
&\quad\; \epsilon_{\texttt{headquarters}}(\#Z, \#2)\,. \\
q_5(\#A, \#Z) \;&:\!-\; \epsilon_{\texttt{capital}}(\#1, \#A), \epsilon_{\texttt{headquarters}}(\#Z, \#2), \\
&\quad\; \phi^{<}_{\texttt{population}}(\#1, \#2)\,. \\
q_6(\#A, \#B, \#Z) \;&:\!-\; \epsilon_{\texttt{capital}}(\#1, \#A), \epsilon_{\texttt{language}}(\#3, \#A), \\
&\quad\; \epsilon_{\texttt{located in}}(\#2, \#B), \epsilon_{\texttt{language}}(\#3, \#B), \\
&\quad\; \epsilon_{\texttt{headquarters}}(\#Z, \#2), \phi^{<}_{\texttt{population}}(\#1, \#2)\,.
\end{aligned}
$$

Interpretability of relational queries. Relational queries are a rather complex type of pattern. They are more easily understood in their graphical representation, allowing to visualize the different objects involved and their connections.

The limitation to monotone conjunctions aims to ensure the interpretability of the queries. First, queries involving both conjunctions and disjunctions would be still more complex and could not be represented as graphs, making interpretation very difficult. Second, because of the heterogeneity of the dataset, negation is equivocal. A predicate might not hold for an object or pair of objects for one of two reasons, either because the attribute is not defined or because it takes a different value. In most cases, the complementary predicate, selecting objects or object pairs for which

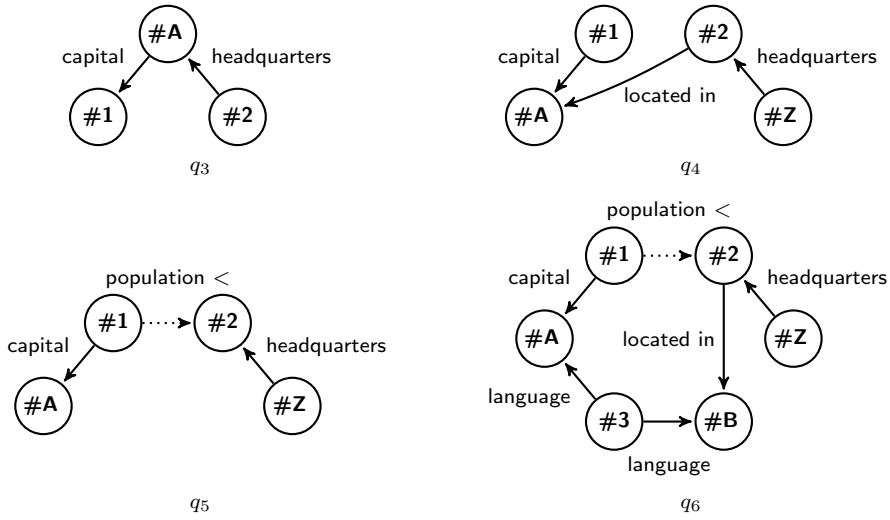


Figure 3.4: Four relational queries.

the attribute is defined but takes a different value or compares differently, can be obtained by replacing the value range by its complement or choosing a different comparison function. For example,

$$\nu^{Republic}_{\texttt{regime}}(\#A) \quad \text{becomes} \quad \nu^{Monarchy}_{\texttt{regime}}(\#A)\ , \text{ and}$$
$$\phi^{<}_{\texttt{population}}(\#1, \#2) \quad \text{becomes} \quad \phi^{\geq}_{\texttt{population}}(\#1, \#2)\ .$$

In addition, we require clauses to be *linked*, meaning that the set of edge predicates connects any two query variables (X, Y). Indeed, relational queries should characterize the connections between the objects of interest and unlinked queries are of little use for this purpose. In particular, query q_5 above does not satisfy this requirement, since the comparison predicate between variables #1 and #2 is not considered for linkage. All other queries are linked.

Furthermore, when mapping the statement onto the data, we require that each variable in the query be matched to a different node in the graph. As common in graph mining, we use subgraph isomorphism or, in terms of logic, θ_{OI}-subsumption [EMS+94], to match queries against the data graph. We consider that the resulting queries are more intuitive. They are also easier to search. For instance, variables #1 and #2 in query q_4 should be mapped to different objects. The substitution

$$\{\#A/\text{US}, \#1/\text{Washington D.C.}, \#2/\text{New-York}, \#Z/\text{UN Security Council}\}$$

complies with this requirement so the pair (US, UN Security Council) supports query q_4. On the other hand, the pair (GB, Commonwealth) does not support this query because London is simultaneously the capital of the United-Kingdom and houses the headquarters of the Commonwealth.

Finally, the definitions above apply to queries of arbitrary arity but our work focuses on queries of arity two, those describing the relations between pairs of objects. While our proposed techniques are tuned towards this type of patterns, they might still be extended to queries of higher arity.

As a word of warning about the terminology, we point out that the term *variable* refers exclusively to a place holder for objects in this introduction and in Article III, while it is used to refer to a vector-valued attribute in the propositional setting described in Articles I and II.

The propositional setting presented in Section 3.1 readily corresponds to the relational case where attributes are restricted to node attributes over a homogeneous set of objects, i.e. such that $\mathcal{A} = \mathcal{N}$, $\mathcal{E} = \emptyset$, and $\mathrm{dom}(A) = \mathcal{O}$, $\forall A \in \mathcal{A}$. In the absence of edge attributes, the connectivity requirement practically restricts the number of variables appearing in the body of the query to one. This variable also necessarily appears in the head as the only query variable. Hence, the support of such a query consists of tuples of size one. This directly maps to propositional monotone conjunctive queries whose support consists of a set of individual objects. In a propositional query, all predicates characterize the same object. For this reason, the variable that refers to that object always remained implicit. In fact, the notation $[A_j \in R_S]$ used in propositional queries is a short hand for

$$\nu_{A_j}^{R_S}(o) = [o \in \mathrm{dom}(A_j) \ \wedge \ A_j(o) \in R_S] \ .$$

This shows that the propositional setting is a restriction of the general relational setting.

Propositional redescription mining is directed towards the description of individual objects, as compared to the characterization of connection patterns between multiple objects in the relational setting. Propositionalization tools aim at turning relational datasets into propositional ones by constructing propositional attributes based on the local connections around individual nodes [KZ09, DEV12]. Redescriptions consisting of queries of arity one, such as q_3, can be obtained by applying propositionalization coupled with a propositional redescription mining algorithm. More generally, as argued in Article III, replacing a fully relational method by propositionalization coupled with a propositional redescription mining algorithm does not allow to maintain the full connectivity information.

Chapter 4

Exploration Strategies

Given a query language, the space of possible queries needs to be explored in search of pairs that constitute good redescriptions. Combined with diverse constraints on the redescriptions, different query languages give rise to different search spaces. Beneficial properties of the language and constraints, such as anti-monotonicity, might allow for a particularly efficient exploration. However, this is not the case in general.

Considering propositional data, there are 2^{2^k} non-equivalent unrestricted Boolean expressions over a set of k predicates, the number of k-place truth functions. Hence, given a set of n predicates, there are

$$\kappa_n = \sum_{k=0}^{n} \binom{n}{k} 2^{2^k}$$

different expressions of arbitrary length. Furthermore, when looking at two datasets with $n_{\mathbf{L}}$ and $n_{\mathbf{R}}$ predicates respectively, there are potentially up to $(\kappa_{n_{\mathbf{L}}} - 1)(\kappa_{n_{\mathbf{R}}} - 1)$ pairs of non-empty queries to examine.

For reasons of interpretability, one would generally only consider queries involving at most a small fixed number of predicates and impose syntactic restrictions on the combination of predicates, significantly reducing the amount of candidate pairs. Still, it is generally too large to allow for an exhaustive enumeration. In the presence of non-Boolean attributes, the number of predicates that can be constructed might be extremely large. Furthermore, in the relational case, there can be infinitely many distinct valid queries. Thus, resorting to heuristics is a practical necessity.

The rest of this chapter outlines and compares three generic exploration strategies for mining redescriptions.

4.1 Query Mining and Pairing

The simplest exploration strategy consists of two steps. First, individual queries are mined from the dataset independently. Second, queries with similar supports are paired to form redescriptions.

On one hand, if the number of views is small, the most practical approach is to mine queries from each view separately, then to pair them across the views. On the other hand, if the number of views is large, for instance when each predicate is associated to a distinct view, one might mine queries over all predicates pooled together, then pair queries with similar supports that involve predicates from disjoints sets of views.

Example 5. *Continuing with our running example, we now consider the case of a propositional dataset divided into two views. Specifically, one view consists of the seven geographic attributes,* $G_1, \ldots, G_7$*, listed in Table 3.1, while the other consists of the four geopolitical attributes of Table 3.2 (a), henceforth denoted as* $P_1, \ldots, P_4$*.*

Faced with such data, the first mining strategy would be carried out by mining geographic and geopolitical queries independently, before pairing them based on support similarity.

The main advantage of such a mine-and-pair scheme is to allow the adaptation of frequent itemset mining algorithms in a very straightforward fashion. Over the last couple of decades, a great number of algorithms have been developed to mine monotone conjunctive queries over a fixed set of propositional predicates [AS94, PCY95, ZH02, CG02, HPYM04], to cite only a few among the most prominent examples. Typically, they exploit the anti-monotonicity of the support of queries to safely prune the search space, resulting in highly efficient complete enumeration procedures.

An alternative to mining and pairing is to replace the second step with a splitting procedure. That is, pool together all predicates for the initial mining step, then split the queries depending on views. However, the existence of a query does not imply that it can be split into two subqueries that both hold with the same supports. More generally, there is no guarantee that there will be a way to split the query found into two subqueries over disjoint views with sufficiently similar supports and even less so with relational queries, when the connectivity needs to be maintained.

When the data originate from two views, monotone conjunctive redescriptions can be mined exhaustively in a level-wise fashion similar to the *Apriori* algorithm [AS94, MTV94]. The support cardinality of both queries and of their intersection, as well as some associated measures, are

antimonotonic and can be used safely for pruning. However, support similarity functions are typically not antimonotonic, even in this simplest case. This strategy is adopted in Article IV, where the search for the best rule at each iteration is carried out exhaustively.

Mining and pairing is best suited for exhaustive search. We now turn to schemes that can be used for an exploration relying on heuristics.

4.2 Alternating Scheme

Another strategy for mining redescriptions is to use an alternating scheme. The general idea is to start with one query, find a good matching query to complete the pair, drop the first query and replace it with a better match, and continue to alternate in this way, constructing a fresh query on either side until no further improvement can be achieved.

For example we would start with an initial query $q_{\mathbf{L}}^{(0)}$ over geographic attributes and look for a good matching query over geopolitical attributes, $q_{\mathbf{R}}^{(1)}$. Next, we would look for another query on the geographic attributes, $q_{\mathbf{L}}^{(2)}$, that forms a better pair $(q_{\mathbf{L}}^{(2)}, q_{\mathbf{R}}^{(1)})$, and so on.

In fact, if one side of the redescription is fixed, finding an optimal query to complete the pair constitutes a binary classification task. The entities supporting the fixed side provide positive examples and the remaining entities might be considered as negative examples. Thus, any feature-based classification technique could potentially make up the basis for a redescription mining algorithm, with the associated query language consisting of the possible classification criteria.

However, to be consistent with our position on the interpretability of queries, we exclude for instance the direct use of linear classifiers. Indeed, the resulting weight vectors have reduced interpretability compared to explicit constraints on the range of attributes.

This alternating scheme was introduced by Ramakrishnan et al. [RKM+04]. Their CARTwheels algorithm is based on decision trees and the query language consists of the resulting rules. In Article III we propose an alternating scheme for mining relational redescriptions. It relies on a relational query miner in order to find matching queries to complement the current pair.

The question of finding good starting points arises naturally. One option is to randomly partition the entities into positive and negative examples, using one or several such partitions to initialize the search, instead of actual queries [RKM+04]. Queries that consist of a single predicate, i.e. the simplest possible queries, offer another choice for the initialization. This

option, adopted in Article III, is particularly appropriate for the relational setting, where the number of possible partitions is extremely large and a majority would not result in any query. In either case, accepting queries that do not match the fixed side very well during the first iterations can help increase the exploratory power of the algorithm.

For a fixed number of starting points and a limit on the number of alternations, the complexity of such a scheme depends primarily on the complexity of the chosen classification algorithm.

4.3 Greedy Atomic Updates

Finally, a third exploration strategy relies on iteratively finding the best atomic update to the current query pair. More precisely, given a pair of queries, one tries to apply atomic operations on either query to improve the candidate redescription, until no further improvement can be achieved. Conceptually, atomic operations at hand include the addition, deletion and edition of predicates. That is, one might add a fresh predicate to the query, remove a predicate from the query or alter some predicate already occurring in the query, in particular, by modifying the range of the truth vale assignment.

For example, if our current candidate redescription is

$$(G_2 \wedge [1 \leq G_6 \leq 10] \ , \ [100 \leq P_4])$$

by adding, deleting and editing a predicate, we might modify it respectively to

$$\begin{aligned} &(G_2 \wedge [1 \leq G_6 \leq 10] \ , \ [100 \leq P_4] \vee [P_3 = \text{Monarchy}]) \ , \\ &\quad ([1 \leq G_6 \leq 10] \ , \ [100 \leq P_4]) \text{ or} \\ &(G_2 \wedge [5 \leq G_6 \leq 10] \ , \ [100 \leq P_4]) \ . \end{aligned}$$

Memorization of the explored queries can be employed to prevent the algorithm from repeating itself. For the initialization, one might consider the pairs of best matching predicates constructed with any two attributes from different views.

This strategy, restricted to addition of predicates, i.e. extensions, was first introduced as the Greedy algorithm by Gallo et al. [GMM08]. Building upon this work, we proposed the ReReMi algorithm in Article I. The algorithm is strengthened with a beam search to keep the current top candidates at each step instead of focusing on the single best improvement.

Typically, such an algorithm would consider in turn each attribute to generate modified candidates, a subset of which will be selected and further

updated at the next step. The running time upper bound for this strategy is in the order of the product of the number of starting points, the maximal number of iterations and the beam width multiplied by the number of real-valued attributes times the squared number of objects plus the number of Boolean attributes and of categories of nominal attributes times the number of objects. For instance, in the example of Section 6.3, this product equals $100 \times 10 \times 4 \times (48 \times 2575^2 + 190 \times 2575) \approx 10^{12}$. In fact, not all objects can affect the support for a particular extension and determining the optimal extension attainable with a given real-valued attribute is quadratic in the number of cut points, which is at most the number of distinct values of the attribute and usually much smaller than the number of rows, as we argue in Article I. Thus, this strategy proved feasible in practice.

When global constraints on the query need to be enforced, like connectivity in the relational case, the alternating scheme presented previously is better suited compared to such atomic updates. On the other hand, atomic extensions might be favored when the construction of individual predicates is costly, as for instance when it involves finding the best interval for real-valued attributes. Indeed, in such cases, building a fresh query from scratch at each alternation can represent a waste of energy if the successive queries are close variations of their replacement and, in effect, the same predicates are generated over and over again.

Chapter 5

Pattern Selection

In this chapter we take a closer look into pattern selection. After having defined query languages and sketched methods for exploring the search space, we now discuss the evaluation of quality and the selection of redescriptions.

First, we consider the definition of quality criteria and their enforcement with respect to individual redescriptions. Such quality criteria, arising from background knowledge or particular domain requirements and modelled as a set of constraints $\mathcal{C}$, determine a bias towards individual redescriptions.

However, the aim of a data mining task generally lies in finding patterns that together describe the data well, instead of finding good patterns taken in isolation. That is, the analyst is interested in identifying a high quality set of patterns rather than a set of high quality patterns. Thus, we also consider the problem of mining sets of redescriptions.

5.1 Individual Patterns

In this section, we inventory criteria that affect the quality of redescriptions, before giving an outline of how they are enforced during the mining process.

5.1.1 Quality Criteria

Soon after the problem of association rule mining was defined [AIS93] and the first efficient solution, the now standard *Apriori* algorithm, was proposed [AS94, MTV94], it became clear that frequency and confidence are not sufficient to ensure the quality of the results [KMR$^+$94, SVA97]. Similarly with redescriptions, while the structural difference in the queries and the similarity of their supports are defining features, they are not sufficient to guarantee the quality of the results. Other crucial aspects need to be taken into account.

The quality of a pattern is a rather abstract property. It results from a combination of characteristics that we try to evaluate with objective criteria. For instance, we might consider that a good redescription is a redescription with easily interpretable queries and statistically significant supports. That said, we still need to define precisely what is meant by interpretability and statistical significance, preferably in an operative way, by defining means to measure these characteristics.

Queries. Besides structural difference, in other words, the requirement that the attributes over which the queries forming a redescription are expressed must belong to distinct views, expressivity and interpretability are the main desirable characteristics for the queries of a redescription. For instance, long and nested formulae are generally hard to interpret, and are therefore of little interest for describing the data. Yet, too strong restrictions imposed on the syntactic complexity of queries might severely limit the expressive power of the language. Hence, a balance needs to be struck between these partly conflicting characteristics, which are moreover difficult to measure. The expressivity of the language and interpretability of individual elements are largely determined by the syntactic restrictions imposed on the construction of statements, discussed in Chapter 3. In addition, a simple means to control the complexity of a query is to limit its length as it is generated.

Support. The similarity between the supports of the queries of a redescription is a defining property of a redescription, also called its accuracy. As mentioned in Section 2.1, the similarity relation $\sim$ is generally specified as a set similarity function together with a threshold. Various functions can be used for this purpose. For a pair of queries $(q_{\mathbf{L}}, q_{\mathbf{R}})$, we denote by $E_{1,1}$, $E_{1,0}$, $E_{0,1}$ and $E_{0,0}$ the subsets of entities that support both queries (i.e. $E_{1,1} = \mathrm{supp}(q_{\mathbf{L}}) \cap \mathrm{supp}(q_{\mathbf{R}})$), support only $q_{\mathbf{L}}$, support only $q_{\mathbf{R}}$ and do not support either queries, respectively. Then, examples of similarity functions include the following:

$$\begin{aligned}
\text{matching number} &= |E_{1,1}| + |E_{0,0}| \ , \\
\text{matching ratio} &= \frac{|E_{1,1}| + |E_{0,0}|}{|E_{1,0}| + |E_{1,1}| + |E_{0,1}| + |E_{0,0}|} \ , \\
\text{Russel \& Rao coefficient} &= \frac{|E_{1,1}|}{|E_{1,0}| + |E_{1,1}| + |E_{0,1}| + |E_{0,0}|} \ , \\
\text{Jaccard coefficient} &= \frac{|E_{1,1}|}{|E_{1,0}| + |E_{1,1}| + |E_{0,1}|} \ ,
\end{aligned}$$

$$\text{Dice coefficient} = \frac{2\,|E_{1,1}|}{|E_{1,0}| + 2\,|E_{1,1}| + |E_{0,1}|}\,, \text{ and}$$

$$\text{Rogers \& Tanimoto coefficient} = \frac{|E_{1,1}| + |E_{0,0}|}{|E_{1,0}| + 2\,|E_{1,1}| + |E_{0,1}| + |E_{0,0}|}\,.$$

The Jaccard coefficient is commonly used in redescription mining. This choice is motivated mainly by the simplicity of the measure and its agreement with the symmetric approach adopted in redescription mining. Indeed, the Jaccard coefficient weights the support of the two queries equally. In addition, it is scaled to the unit interval without involving the set of entities that support neither queries, $E_{0,0}$. This is an asset, particularly in the relational setting, when the dataset is heterogeneous or requires the use of the *open world assumption*, i.e. the assumption that a relation may exist despite not being recorded in the dataset, so that this set may not be easily and appropriately defined.

Besides accuracy, it can be desirable to fix lower or upper bounds on the support cardinality of the queries and possibly on that of the individual predicates involved as well. Also, in the relational case where entities consist of object tuples, more complex constraints can be imposed, for instance on the number of distinct objects appearing at any given position or on the number of distinct tuples up to reordering. These constitute secondary constraints on the redescriptions that might arise from the domain knowledge and help select redescriptions of interest.

Statistical significance. A crucial requirement for the redescriptions mined is that they be statistically significant. To provide new insight about the data at hand, a redescription should not be likely to arise at random from the underlying data distribution. In particular, the accuracy of a redescription should not be readily deducible from the support of its queries. For instance, if both queries cover almost all objects, the overlap of the supports is necessarily large, too, and a high accuracy is no surprise.

Hence, one way to measure the significance of a redescription is to estimate how likely such a pattern is to arise randomly. That is, the presence of the redescription is tested against the null-model where the two queries are assumed to be independent. Consider two statistically independent random queries whose marginal probabilities correspond to those of the queries under consideration. In other words, their marginal probabilities equal the fraction of covered entities $p_{\mathbf{L}} = |\mathrm{supp}(q_{\mathbf{R}})| \,/\, |E|$ and $p_{\mathbf{R}} = |\mathrm{supp}(q_{\mathbf{R}})| \,/\, |E|$, respectively. A p-value representing the probability that these independent queries have an overlap equal to or larger than the one observed can be

computed using the binomial distribution as follows

$$\mathrm{pvalM}(q_{\mathbf{L}}, q_{\mathbf{R}}) = \sum_{s=|E_{1,1}|}^{|E|} \binom{|E|}{s} (p_{\mathbf{L}} p_{\mathbf{R}})^s (1 - p_{\mathbf{L}} p_{\mathbf{R}})^{|E|-s} \ .$$

This is the probability of obtaining a set of same cardinality $|E_{1,1}|$ or larger if each element of a set of size $|E|$ has a probability equal to the product of marginals $p_{\mathbf{L}}$ and $p_{\mathbf{R}}$ to be selected, in accordance with the independence assumption.

Alternatively, a p-value can be computed as the probability that two sets of cardinalities $|\mathrm{supp}(q_{\mathbf{R}})|$ and $|\mathrm{supp}(q_{\mathbf{L}})|$, respectively, drawn independently at random from a set of size $|E|$ have an overlap of cardinality $|E_{1,1}|$ or larger. This is Fisher's exact one-sided p-value [Fis38], evaluated using the hypergeometric distribution:

$$\mathrm{pvalO}(q_{\mathbf{L}}, q_{\mathbf{R}}) = \sum_{s=|E_{1,1}|}^{|E|} \frac{\binom{|\mathrm{supp}(q_{\mathbf{L}})|}{s} \binom{|E|-|\mathrm{supp}(q_{\mathbf{L}})|}{|\mathrm{supp}(q_{\mathbf{R}})|-s}}{\binom{|E|}{|\mathrm{supp}(q_{\mathbf{R}})|}} \ .$$

High p-values indicate that the independence assumption, i.e. the null hypothesis, cannot be rejected and the redescription is then considered less significant. The computation of such theoretical p-values relies on assumptions about the underlying data distribution. Both tests assume that all elements of the population can be sampled with equal probability, from a fixed distribution. The sampling distribution is calculated only on expectation in the former case, while the latter relies on the stronger assumption of fixed marginals. However, the real data might deviate from these simple assumptions, weakening the significance tests.

Theoretical p-values can be complemented by empirical statistical tests, carried out after randomizing the original data. Both approaches rely on statistical hypothesis testing. Developing a well-founded methodology based on this theory to assess the significance of redescriptions requires to consider a number of issues such as appropriate multiple testing with scaling and corrections, as well as property-preserving randomization and uniform sampling of datasets in the case of randomization tests.

These questions do not constitute the core of our contribution and we do not discuss them in depth here. Instead we refer the interested reader to the relevant literature [LR05, Edg95] for general considerations about statistical hypothesis testing and randomization tests or concerning their application to data mining, from the early study of statistical significance of association rules [BMS97, MS98] to recent developments [Oja11, Han12, Vuo12], among others [Web07, ZPT04].

In Article I, we assess the significance of propositional redescriptions using both approaches. Empirical p-values, in particular, were obtained following the approach of Gionis et al. [GMMT07]. Specifically, copies of the original data are generated and randomized so as to maintain, at least approximately, the row and columns marginals. Then, the mining process is run anew on each of the copies. Redescriptions from the original data that are more accurate than a chosen fraction of the redescriptions obtained from the randomized copies are deemed significant with respect to the preserved properties, others are discarded.

We did not study this aspect in the context of relational datasets. Evaluating the statistical significance of complex connection patterns such as our relational queries is a difficult but interesting question. It is open for future investigations, possibly building on works by Hanhijärvi et al. [HGP09] and by Günnemann et al. [GDJE12].

5.1.2 Constraint-based Mining

In the previous section, we discussed characteristics that impact the quality of a redescription and the associated evaluation criteria. Such criteria result in a set of constraints $\mathcal{C}$ that limits the space of acceptable redescriptions. They can be enforced either during the exploration, by pruning the search space, or as a post-processing step, by filtering the output. Clearly, it is preferable to push the constraints as deeply as possible into the search algorithm, as this improves efficiency by preventing the generation of redescriptions only to discard them later on.

Significant effort has been directed towards the integration of various constraints into exhaustive search algorithms [SVA97, GR00], giving rise to *constraint-based data mining*. This integration relies on the classification of constraints according to properties such as anti-monotonicity and succinctness [NLHP98] that determine their behavior and consequently how they should be used to prune the search space safely but optimally.

However, these works focus on conjunctive query languages and exhaustive pattern enumeration, by extending the *Apriori* algorithm. Allowing disjunctions makes these methods inapplicable because anti-monotonicity no longer holds. Then, heuristic approaches are preferred. Beam-search algorithms, in particular, depend on a ranking function to determine the top candidates that will be explored at the next step. Designing an appropriate score to ensure a satisfactory exploration of the search space with respect to a set of quality constraints is far from trivial.

Conceptually, constraint-based data mining is a step towards *inductive databases* [IM96], a framework for data mining where databases in addi-

tion to the usual data also contain patterns over this data [BKM99, DR02, BDRM05]. Inspired by the success of Codd's model [Cod70] and the powerful closure property, this framework proposes to see data mining as the manipulation of patterns using a set of expressive operations, similarly to the way ordinary database records can be manipulated using relational algebra.

Consider the following abstract model of data mining introduced by Mannila and Toivonen [MT97]. Given a language of patterns $\mathcal{L}$, a dataset $\mathcal{D}$ and a selection predicate $\mathcal{S}$, a data mining task aims at determining the theory of $\mathcal{D}$ with respect to $\mathcal{L}$ and $\mathcal{S}$,

$$Th(\mathcal{D}, \mathcal{L}, \mathcal{S}) = \{\phi \in \mathcal{L}\,,\, \mathcal{S}(\phi, \mathcal{D})\}\,.$$

From the point of view of inductive databases, the computation of $Th(\mathcal{D}, \mathcal{L}, \mathcal{S})$ is a generic database operation of evaluating $\mathcal{S}$, known in this context as an inductive query. Redescription mining naturally integrates into this framework. The language of patterns consists of all query pairs, i.e. $\mathcal{L} = Q \times Q$, and the selection predicate takes the form of the accuracy and structural difference requirements together with the auxiliary constraints on the patterns.

Recently, constraint programming has been proposed as a declarative approach for constraint-based data mining [DRGN08, GNZDR11, KBC10]. It shares with inductive databases the aim of providing a generic language for specifying desirable characteristics of patterns, independently of the procedure used to identify the actual patterns. Constraint programming has been applied to redescription mining restricted to monotone conjunctions [GNDR13]. In general, this approach is currently unable to deal efficiently with certain classes of patterns and constraints, but promising for others.

5.1.3 Interactive Data Mining

Selecting patterns by explicitly specifying a set of desired characteristics and the associated means of evaluation offers much flexibility. However, such an ad-hoc approach also has its drawbacks. It might require extensive background knowledge and multiple rounds of trial and errors to familiarize with the tool and tune the parameters so as to obtain good results.

There, an interactive interface that allows the analyst to inspect patterns as they are generated and that readily provides feedback comes in handy. The Siren interface presented in Article II is a first step in this direction. Potentially, by modifying the selection criteria, the analyst is

able to specify his interest dynamically, in response to the output produced hitherto by the mining algorithm.

However, such high flexibility and adaptability might actually enable the analyst to fine-tune the mining process to obtain only the results that confirm his expectations, putting the discovery of new knowledge in jeopardy.

5.2 Sets of Patterns

Even with strict quality requirements, the returned set of redescriptions might be large and contain near duplicates.

For instance, a collection of a dozen results which are minor variations of each other is an undesirable result, even if each redescription is highly accurate and of good quality when considered separately. The problem here lies in the redundancy of the redescriptions. When each of them conveys more or less the same information, communicating the whole set of patterns to the analyst represents a large cognitive overhead compared to returning only one, while the informative content remains almost unchanged.

One way of measuring the redundancy between two redescriptions is to compare the occurring attributes and covered entities, since they carry most of the information of a redescription. In particular, given a set of redescriptions, one can consider the similarity of their attribute sets and support intersection separately or compare the overlap of the area defined by the rows and columns involved in either propositional queries. Still, this is a rather crude way of measuring redundancy.

Overwhelming results are a major issue of data mining algorithms. In the domain of frequent itemset mining, it has been proposed to look for concise representations of the results, that is, to identify a small set of patterns from which the rest can be derived exactly or approximately. Such summaries also provide a condensed representation of the data [MT96]. This leads to the notions of closed itemsets [PBTL99] and free-sets [BB00], among others (see [CRB04] for an overview).

More generally, given a dataset, a typical aim is to mine a small set of patterns that together describe the data well, rather than consider patterns taken in isolation. One approach to pattern set mining is to employ constraint-based techniques as discussed in Section 5.1.2, this time also taking into account constraints on the entire set of patterns such as support overlap or coverage [GNZDR11].

Methods rooted in Information Theory constitute more holistic approaches to selecting sets of patterns. In particular, alternatives based

on compression and on subjective interestingness, are presented in the following sections, respectively.

5.2.1 Compression-based Model Selection

Compression-based approaches for pattern set mining use compression as a selection criterion. They are motivated by the intuition that the data can be compressed more efficiently by exploiting its internal structure, so that uncovering more of the structure results in improved compression. Simultaneously, redundancies among the patterns result in increased compressed size and are therefore penalized.

Different techniques have been studied to select a model for a given dataset based on information theoretic principles such as the Minimum Description Length (MDL) [Grü07] or the Information Bottleneck (IB) [TPB00]. The MDL and IB approaches differ notably in the fact that the first requires a lossless compression scheme while the latter allows for lossy compression.

The central ingredient of the MDL recipe is the definition of an encoding scheme for the patterns. Then, patterns mined from the data can be stored in a table together with their associated code-words and used to encode the data. The aim is to find a set of patterns that yields the shortest encoding of the data while keeping the size of the code table minimal. A prime example of mining tool based on the MDL principle is the Krimp itemsets miner [VvLS11].

Inspired by this approach, we propose a method for mining associations from two-view datasets, or, roughly speaking, for compressing the mapping between datasets using redescriptions, presented in Article IV.

Consider a pair of queries, $(q_{\mathbf{L}}, q_{\mathbf{R}})$, with very similar supports, i.e. an accurate redescription. The fact that $q_{\mathbf{L}}$ holds implies that $q_{\mathbf{R}}$ is very likely to hold too, and vice versa. Therefore, such patterns provide information about the associations between the two views of the dataset and can be used to encode one view given the other. In other words, they allow to translate one view into the other, and we call them translation rules. Then, we look for a set of such rules that together capture the cross-view structure of the data well, as measured by their ability to compress it.

Pairs of propositional monotone conjunctive queries constitute our translation rules and we allow both unidirectional and bidirectional associations, as this provides more flexibility to capture the structure of the dataset and consequently increases the compression ability of our model. More precisely, we consider query pairs where the support of the former query is almost a subset of the support of the latter, i.e. such that the

presence of one query implies the presence of the other, but the converse need not be true.

To summarize, our proposed algorithm identifies pairs of monotone conjunctions which allow to encode one view of the data given the other, or vice versa. Practically, this results in a parameter-free method for mining pairs of queries.

5.2.2 Subjective Interestingness

All the approaches presented so far are concerned only with objective qualities of the patterns, in the sense that the quality depends only on the data and not on the beliefs or preconceived, possibly erroneous, understanding that the analyst possesses prior to the data mining task.

Early on, Silberschatz and Tuzhilin argued that the interestingness of patterns should be evaluated from the point of view of the user [ST95]. They proposed two subjective measures of interestingness. First, actionability depends on whether the analyst can react to the information provided [PSM94]. Second, unexpectedness depends on whether the information surprises the analyst, that is, whether the pattern contradicts the expectations of the analyst, formalized as a system of beliefs [PT98, PT00].

While such approaches arguably employ an extremely simplified representation of the analyst's expectations, they attempt to take these beliefs explicitly into account in the mining process. Therefore, they are called subjective, in contrast to other, objective, approaches.

Already a decade ago, Mannila [Man00] advocated the definition of a theoretical framework for data mining, arguing for the usefulness of such formalization and suggesting five possible candidates for the role, including inductive databases and data-mining as data compression. Pursuing this endeavor, De Bie [DB11a] recently proposed a subjective information theoretic framework for data mining. It is based on the idea that the data mining task can be considered as an exchange of information between the mining process and the analyst.

From this point of view, the data analyst has initial apriori beliefs about the data, modelled as a distribution over possible datasets. During the mining process, information about the data is communicated to the analyst in the form of patterns, allowing him to adjust his beliefs. Then, the amount of new information conveyed by a pattern, i.e. its subjective interestingness, is measured as the reduction of the uncertainty in the data miner's beliefs.

Significance testing approaches based on data randomization mentioned in Section 5.1.1 share some similarities with this line of work. The apriori knowledge of the analyst consists of the preserved properties, so that his

belief is modelled by sampling datasets that possess such properties. However, these empirical approaches are less scalable and suffer from limited resolution compared to the analytical alternative [DB11b]. In addition, they do not allow to model belief updates.

Relying on strong roots in information theory, this framework provides a principled way to define the subjective quality of patterns, as well as the cost of their transmission, i.e. their description length. Formalizing the mining process within this framework should allow to adapt existing algorithms and design new ones so as to maximize the transfer of information from the data to the analyst, for various families of patterns. In particular, integrating redescription mining into this framework is an attractive direction for future research.

Chapter 6

Illustrated Discussion

This chapter provides a practical illustration of the redescription mining task. We present examples of redescriptions mined with the different algorithms we developed, from datasets of diverse domains and using various query languages. These examples complement those of the original publications.

In a sense, the present chapter is a showcase for redescription mining. It is intended to exhibit the power of the method, its versatility, expressivity and interpretability, and not to constitute an experimental evaluation. Detailed documented assessments of the proposed algorithms can be found in the corresponding original publications.

Simultaneously, this exposition provides a basis for a critical discussion of redescription mining. Indeed, through these examples, we point out some weaknesses and drawbacks of the method, which could benefit from further investigations.

We start with a summary of the different algorithms proposed in this thesis with which the results illustrating this chapter were obtained. Each of the four proposed algorithms, indicated in bold in the text, combines aspects of redescription mining discussed in the previous chapters, as outlined below.

6.1 Overview of the Algorithms

As our main contribution, we extended redescription mining outside the world of propositional queries over Boolean attributes. We studied more general query languages and associated algorithms, making the task applicable to a broader range of domains and problems.

In Article I, we proposed the **ReReMi** algorithm for propositional re-

description mining. Specifically, the query language considered consists of propositional linearly parsable queries (Section 3.1.2) over Boolean, nominal and real-valued predicates (Section 3.1.1). Our algorithm was built upon the Greedy algorithm [GMM08], and similarly constructs queries by successive atomic extensions (Section 4.3). Compared to its predecessor, it can handle non-Boolean attributes and missing values and uses a beam search to maintain top candidates, improving exploration. The search is primarily driven by the Jaccard coefficient as the support similarity function (Section 5.1.1). Auxiliary constraints on the redescriptions are enforced ad-hoc by means of ranking and filtering (Section 5.1.2). In addition, we use randomization methods to assess the statistical significance of the obtained redescriptions (Section 5.1.1).

In Article III, we proposed an algorithm for relational redescription mining, called **Arrm**. Node and edge predicates are built over Boolean and nominal object and relation attributes, respectively, while comparison predicates are obtained from real-valued object attributes (Section 3.2.1). These three types of predicates are then combined together into relational queries (Section 3.2.2). To explore the space of queries, we resort to an alternating scheme (Section 4.2). We used the Jaccard coefficient as our measure of choice for accuracy, but it can easily be replaced with another set similarity function (Section 5.1.1). The generation of non-compliant candidates with respect to quality constraints is prevented whenever possible, and filtering is applied to the output to exclude remaining low-quality results (Section 5.1.2).

As we argue in Chapter 4, exploring the space of query pairs with iterative atomic updates is best suited to the propositional setting in the presence of real-valued attributes as it reduces the need for computationally intensive on-the-fly discretization. This approach naturally maps to linearly parsable queries. In the propositional setting, connectivity represents a global constraint on the queries that makes the alternating scheme the most appropriate exploration strategy.

The investigation of pattern selection methods, focusing on the example case of redescriptions, is our second major contribution.

First, we developed an interface for visualizing and mining propositional geospatial redescriptions (Section 3.1), presented in Article II. The proposed interface, called **Siren**, relies on the ReReMi algorithm as its core component. With this tool we take a first step towards interactive and instant redescription mining. Ultimately, such an endeavor could support an entirely interactive selection of redescriptions, by allowing to visualize

results as they are generated and, in response, adjust the parameters of the running algorithm (Section 5.1.3). By giving the analyst as much control as possible over the mining algorithm and filtering procedures, it provides a manually adjustable solution to the selection problem at hand.

Second, we proposed a compression-based method for mining small sets of directional associations from two-view datasets, which can be understood as a preliminary method for mining sets of redescriptions (Section 5.2.1). Specifically, the aim is to find a set of patterns that best describes one side of the data given the other side and vice versa. In other words, we seek to translate one side into the other and hence call such patterns *translation rules*. The algorithm to find them, dubbed **Translator**, is presented in Article IV. This algorithm uses exhaustive search with pruning (Section 4.1) and is limited to propositional monotone conjunctions (Section 3.1.2).

Both approaches have advantages and drawbacks. The first approach is very flexible but lacks a theoretical basis, while the second approach constitutes a principled method rooted in information theory but is currently applicable only to a very restricted query language and does not allow to incorporate background knowledge.

The redescriptions presented in the rest of this chapter are sampled from larger sets of results. The indices appearing in the first column of the tables of examples stand for the position of the corresponding redescriptions in the entire result set ordered by decreasing Jaccard coefficient. In the text, we use the table reference together with this index to refer to a redescription, e.g. we refer to the second redescription in Table 6.2 as redescription 6.1(26). For each redescription, we indicate the right-hand side and left-hand side queries, denoted by $q_{\mathbf{L}}$ and $q_{\mathbf{R}}$, respectively, as well as the Jaccard coefficient, J, and the cardinality of the support intersection, $|E_{1,1}|$.

6.2 *Computer Science Bibliography*

The first illustration concerns publication patterns in computer science research.

Dataset. Specifically, the dataset was obtained from the DBLP Computer Science Bibliography data base.[1] It consists of a pair of matrices with authors as the objects. The first matrix defines the venues in which each author has published, while the second defines other authors with whom they have published. $\texttt{DBLP}_{\texttt{F}}$ is a dataset with 6455 authors and 304

[1] Data retrieved from `http://dblp.uni-trier.de/db` in March 2010.

Table 6.1: Sample of redescriptions from $\texttt{DBLP}_{\texttt{FB}}$ mined with ReReMi.

$q_{\mathbf{L}}$	$q_{\mathbf{R}}$	J	$\lvert E_{1,1}\rvert$
(1) ICDM ∧ CIKM ∧ APWeb ∧ SIGIR	Q. Yang ∧ W. Fan	0.714	10
(26) CCCG ∧ SODA ∧ GD	M. Yvinec ∨ K. Kriegel ∨ J. O'Rourke	0.409	27
(27) VLDB ∧ SDM ∧ SIGMOD ∧ KDD	J. Han ∧ P. S. Yu	0.407	11
(36) CCCG ∧ SODA ∧ SoCG	K. Kriegel ∨ O. Devillers ∨ K. L. Clarkson ∨ D. M. Mount	0.383	49
(38) EUROCRYPT ∧ CRYPTO	S. Halevi ∨ U. M. Maurer ∨ Y. Desmedt ∨ D. Naccache	0.382	58
(56) SDM ∧ SIGMOD ∧ ICDE ∧ KDD	(H. Mannila ∨ P. S. Yu) ∧ J. Han	0.367	11
(59) COLT ∧ ICML	A. J. Smola ∨ R. Khardon ∨ S. P. Singh ∨ Y. Singer	0.366	34
(60) STOC ∧ EUROCRYPT ∧ CRYPTO	R. Ostrovsky ∨ P. Landrock	0.365	35
(71) PODC ∧ STOC ∧ EUROCRYPT	(K. Kurosawa ∨ A. Sahai) ∧ R. Canetti	0.351	13
(72) SODA ∧ SoCG ∧ WADS	S. Bereg ∨ F. P. Preparata ∨ E. D. Demaine	0.351	54
(92) FOCS ∧ STOC ∧ EUROCRYPT ∧ CRYPTO	R. Ostrovsky	0.342	26

Table 6.2: Sample of redescriptions from $\texttt{DBLP}_{\texttt{F}}$ mined with ReReMi.

$q_{\mathbf{L}}$	$q_{\mathbf{R}}$	J	$\lvert E_{1,1}\rvert$
(1) [1 ≤ SEBD ≤ 8] ∧ [1 ≤ LPNMR] ∧ [1 ≤ SIGMOD ≤ 2]	([1 ≤ M. Lenzerini] ∨ [1 ≤ F. Giannotti]) ∧ [2 ≤ N. Leone]	0.909	10
(10) [1 ≤ ICDM≤ 12] ∧ [1 ≤ CIKM ≤ 5] ∧ [1 ≤ APWeb ≤ 10] ∧ [1 ≤ SIGIR]	([1 ≤ J. Xu] ∨ [1 ≤ B. Zhang]) ∧ [1 ≤ W. Fan ≤ 9]	0.667	10
(33) [7 ≤ CCCG ≤ 22] ∧ [2 ≤ SoCG ≤ 9]	[2 ≤ M. H. Overmars] ∧ [4 ≤ E. D. Demaine]	0.524	11
(42) [1 ≤ ICDM] ∧ [1 ≤ DASFAA ≤ 10] ∧ [1 ≤ WAIM] ∧ [1 ≤ SIGMOD]	([4 ≤ J. Pei] ∨ [1 ≤ L. Zhang] ∨ [1 ≤ G. Yu]) ∧ [1 ≤ J. Han]	0.500	10
(48) [1 ≤ ESA ≤ 3] ∧ [7 ≤ GD]	[1 ≤ F-J. Brandenburg]	0.476	10
(49) [1 ≤ NIPS ≤ 20] ∧ [10 ≤ COLT]	[1 ≤ N. Cesa-Bianchi ≤ 2]	0.476	10
(57) [5 ≤ FOCS] ∧ [2 ≤ STOC] ∧ [4 ≤ CRYPTO ≤ 28]	[1 ≤ O. Goldreich] ∧ [1 ≤ S. Micali]	0.467	14
(70) [1 ≤ PODC ≤ 9] ∧ [2 ≤ CRYPTO] ∧[1 ≤ STOC]∧[2 ≤ EUROCRYPT]	([1 ≤ R. Venkatesan] ∨ [1 ≤ R. Gennaro]) ∧ [1 ≤ R. Ostrovsky]	0.455	15
(72) [2 ≤ CRYPTO] ∧ [2 ≤ STOC ≤ 23] ∧[2 ≤ EUROCRYPT]∧[2 ≤ FOCS]	[1 ≤ R. Ostrovsky] ∧ [1 ≤ R. Canetti]	0.452	14
(84) [2 ≤ CRYPTO ≤ 10] ∧ [2 ≤ EUROCRYPT ≤ 12]	[1 ≤ R. Gennaro] ∨ [1 ≤ E. F. Brickell ≤ 2] ∨ [1 ≤ V. Rijmen]	0.435	37
(98) [4 ≤ SEBD ≤ 12]	[2 ≤ S. Paraboschi] ∨ [1 ≤ F. Mandreoli] ∨ [1 ≤ G. Greco]	0.431	31

Table 6.3: Glossary of computer science venues.

Acronym	Venue
APWeb	Asia-Pacific Web Conference
CCCG	Canadian Conference on Computational Geometry
CIKM	International Conference on Information and Knowledge Management
COLT	Computational Learning Theory
CRYPTO	International Cryptology Conference
DASFAA	Database Systems for Advanced Applications
ESA	European Symposium on Algorithms
EUROCRYPT	Int. Conference on the Theory and Applications of Cryptographic Techniques
FOCS	IEEE Annual Symposium on Foundations of Computer Science
GD	Graph Drawing
ICDE	International Conference on Data Engineering
ICDM	IEEE International Conference on Data Mining
ICML	International Conference on Machine Learning
KDD	Knowledge Discovery and Data Mining
LPNMR	Logic Programming and Non-Monotonic Reasoning
NIPS	Neural Information Processing Systems
PODC	ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing
SDM	SIAM International Conference on Data Mining
SEBD	Sistemi Evoluti per Basi di Dati (Italian Symp. on Advanced DB Systems)
SIGIR	Int. Conference on Research and Development in Information Retrieval
SIGMOD	ACM SIGMOD Conference
SoCG	Symposium on Computational Geometry
SODA	Symposium on Discrete Algorithms
STOC	Symposium on the Theory of Computing
VLDB	Very Large Data Bases Conference
WADS	Workshop on Algorithms and Data Structures
WAIM	International Conference on Web-Age Information Management

conferences containing information on how many times each author has published in each venue and with each other author. We denote as $\texttt{DBLP}_{\texttt{FB}}$ the Boolean version of this dataset, that is, the dataset obtained by turning every positive value to one.

Results. A sample of redescriptions mined with the ReReMi algorithm from the Boolean and numerical versions of the DBLP dataset are shown in Tables 6.1 and 6.2 respectively. A glossary of the venues appearing in the redescriptions is provided in Table 6.3. In this context, a redescription consists of a pair of queries over venues and coauthors, respectively, obtained by combining non-negated predicates into linearly parsable queries using conjunctions and disjunctions. The two settings differ solely in the type of predicates used, Boolean in the first case, real-valued in the second.

The redescriptions obtained identify subareas of computer science such as machine learning or cryptology, characterized by prime conferences and key researchers of the respective fields. For instance, redescription 6.2(33) characterizes eleven researchers having multiple publications at SoCG and

CCCG, i.e. contributing to both these major computational geometry conferences, and having collaborated with either Prof. Mark Overmars or Prof. Erik D. Demaine, two key researchers in that area.

More generally, when analyzing bibliographic data, redescription mining can shed light on the communities of researchers that make up the field, arranged by specialty area, complementing other approaches to publication network and scientific collaboration analysis [New01a, New01b, MBKN13].

Discussion. The redescriptions found with either setting share strong similarities. The use of actual counts of publications rather than Boolean indicators allows for finer tuning of the queries. This achieves higher accuracies but also results in multiple redescriptions with support at the acceptability threshold. In this example, the minimum support threshold was set to 10. In the real-valued setting we obtained many redescriptions with support exactly 10, unlike in the Boolean setting. This points to a greater sensibility of the algorithm to such thresholds, due to its increased capacity to adjust the queries, which needs to be controlled to prevent the generation of spurious results.

Because an exhaustive exploration of the space of queries is not feasible, our algorithms rely on heuristics for finding the top redescriptions. In Article I, we present experiments with synthetic data showing that the ReReMi algorithm is able to recover planted redescriptions. However, despite this empirical evidence, the approach is not guaranteed to find the strongest patterns in general.

6.3 *Bioclimatic Niches*

As a second illustration, we consider an application of redescription mining in the domain of biology, namely, to find bioclimatic envelopes. In biology, the bioclimatic constraints that must be met for a certain species to survive constitute that species' bioclimatic envelope or niche (here restricted only to environmental variables in the Grinnellian sense of the term [Gri17], not inter-species competition or such).

Dataset. We consider a dataset, denoted as `Bio`, characterizing the climate and fauna of Europe. Our objects consist of spatial areas of Europe, roughly squares of 50 km sides.[2] The data itself is composed from two publicly available data bases: the European mammal atlas [MJAB+99] and the Worldclim climate data [HCP+05]. The mammals data contains

[2]For details of the grid see `www.luomus.fi/english/botany/afe/index.html`.

Table 6.4: Sample of redescriptions from `Bio` mined with ReReMi. $t_X^{\min}, t_X^{\max}$, and t_X^{avg} stand for minimum, maximum, and average temperature of month X in degrees Celsius, and p_X^{avg} stands for average precipitation of month X in millimeters.

	$q_{\mathbf{L}}$	$q_{\mathbf{R}}$	J	$\lvert E_{1,1}\rvert$
(3)	Polar bear	$[-4.5 \leq t_{\text{Oct}}^{\max} \leq -1.0]$	0.973	36
(4)	Polar bear	$[1.0 \leq t_{\text{Sep}}^{\max} \leq 3.5]$	0.973	36
(7)	Wood mouse ∨ Azores Noctule	$\big(([3.0 \leq t_{\text{Mar}}^{\max}] \wedge [9.8 \leq t_{\text{Oct}}^{\max}]) \vee [9.7 \leq t_{\text{Jul}}^{\max} \leq 14.0]\big) \wedge [0.4765 \leq t_{\text{Oct}}^{\text{avg}} \leq 19.5860]$	0.842	1703
(9)	Bank Vole ∨ Steppe Mouse ∨ Northern Red-backed Vole ∨ Harbor Seal	$[-9.2 \leq t_{\text{Dec}}^{\max} \leq 12.8000] \wedge [7.1556 \leq t_{\text{Aug}}^{\text{avg}} \leq 23.089] \wedge [34.714 \leq p_{\text{Jun}}^{\text{avg}}] \wedge [47.625 \leq p_{\text{Aug}}^{\text{avg}}]$	0.838	1696
(14)	Wood mouse	$\big(([3.0 \leq t_{\text{Mar}}^{\max}] \wedge [4.2 \leq t_{\text{Nov}}^{\max}]) \vee [9.7 \leq t_{\text{Jul}}^{\max} \leq 13.2]\big) \wedge [-5.4944 \leq t_{\text{Dec}}^{\text{avg}} \leq 13.133]$	0.828	1685
(23)	Cape Hare ∨ European Hare ∨ Algerian Mouse	$([15.208 \leq t_{\text{Jul}}^{\text{avg}} \leq 26.36] \wedge [-12.9 \leq t_{\text{Dec}}^{\min} \leq 8.9]) \vee [10.4 \leq t_{\text{Sep}}^{\text{avg}} \leq 12.187] \vee [112.75 \leq p_{\text{Apr}}^{\text{avg}}]$	0.808	1677
(30)	Mountain Hare	$([t_{\text{Sep}}^{\text{avg}} \leq 12.992] \wedge [7.6 \leq t_{\text{Sep}}^{\max} \leq 17.2] \wedge [13.5 \leq t_{\text{Jul}}^{\max} \leq 22.5]) \vee [81.111 \leq p_{\text{Apr}}^{\text{avg}} \leq 81.222]$	0.782	688
(39)	Balkan Snow Vole ∨ Field Vole ∨ Azores Noctule	$[11.5 \leq t_{\text{Jun}}^{\max} \leq 24.5] \wedge [12.2 \leq t_{\text{Jul}}^{\max} \leq 26.7] \wedge [34.714 \leq p_{\text{Jun}}^{\text{avg}} \leq 175.0] \wedge [42.0 \leq p_{\text{Sep}}^{\text{avg}} \leq 183.06]$	0.751	1343
(54)	Harvest Mouse ∧ European Mole	$[-0.3 \leq t_{\text{Apr}}^{\min} \leq 8.8] \wedge [19.4 \leq t_{\text{Aug}}^{\max} \leq 27.2] \wedge [45.417 \leq p_{\text{Jun}}^{\text{avg}}] \wedge [48.75 \leq p_{\text{Aug}}^{\text{avg}} \leq 126.56]$	0.677	774
(56)	(Daubenton's Bat ∧ Eurasian Pygmy Shrew) ∨ Balkan Snow Vole	$([t_{\text{Nov}}^{\min} \leq 6.2] \wedge [14.0 \leq t_{\text{May}}^{\max} \leq 20.6] \wedge [48.75 \leq p_{\text{Aug}}^{\text{avg}} \leq 165.6500]) \vee [1.025 \leq t_{\text{Apr}}^{\text{avg}} \leq 1.0917]$	0.669	870

presence/absence information of mammal species in Europe, and the aggregated climate data contains minimum, average, and maximum monthly temperatures as well as average monthly precipitation.

Results. Table 6.4 presents redescriptions mined from this dataset with the ReReMi algorithm. Since the objects considered in this task correspond to geographic locations, the redescriptions can be naturally plotted on maps. The maps generated with the Siren interface for these sample results are shown in Figure 6.1.

These redescriptions accurately characterize areas, often contiguous, that share similar climatic conditions and constitute the habitat of particular species. For instance, an area spreading from the Pyrenees to the Baltic states is described in redescription 6.4(54) as the region where the harvest mouse and the European mole cohabit and where a conjunction of temperatures and precipitation conditions is encountered.

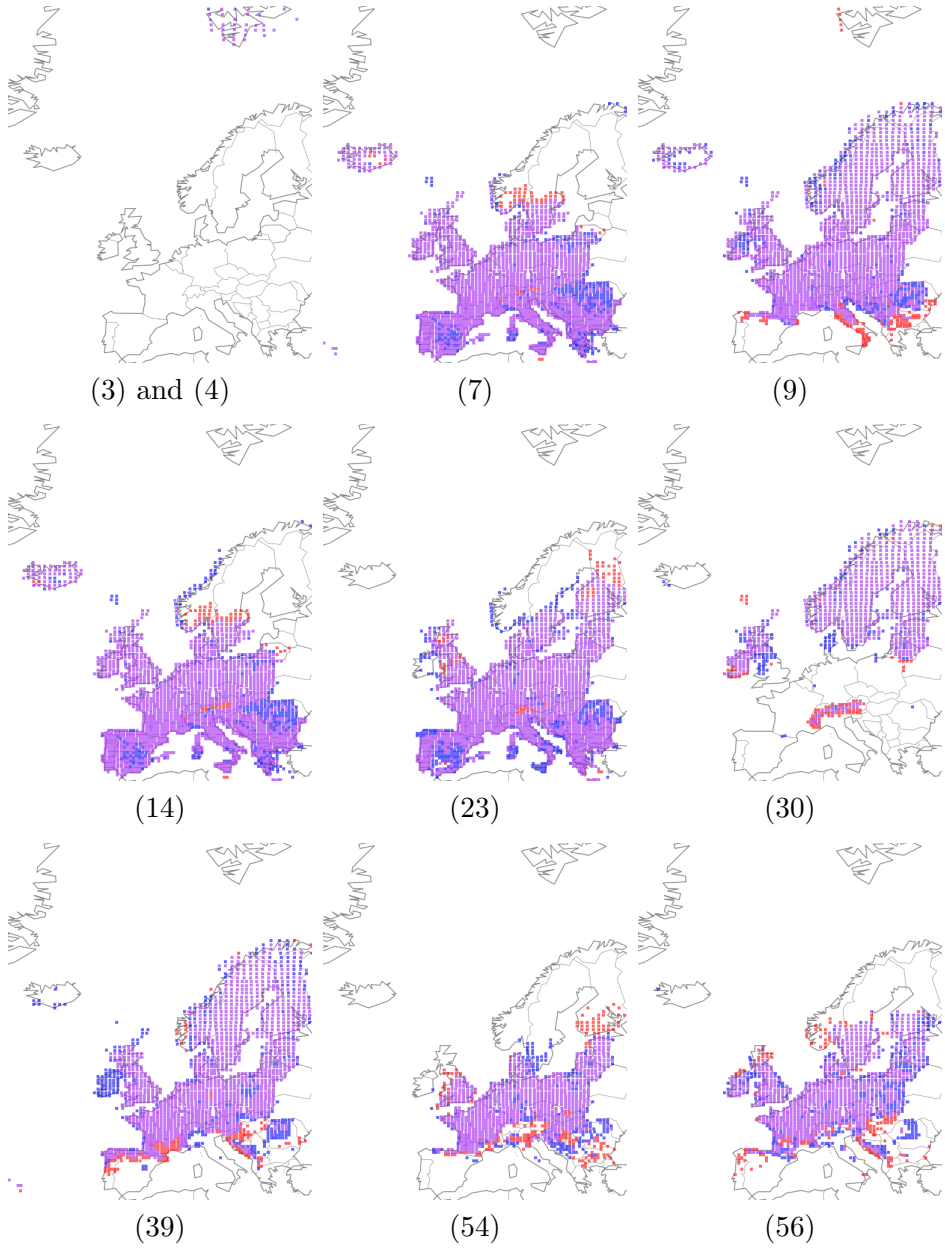


Figure 6.1: Support of the redescriptions on `Bio` shown in Table 6.4. For each redescription, purple, red and blue squares indicate areas where both queries hold ($E_{1,1}$), only the left query holds ($E_{1,0}$) and only the right query holds ($E_{0,1}$), respectively.

Redescriptions of this sort define the bioclimatic niche of species taken individually or in combination. Finding such niches is an important problem in biology that, for instance, can help predict the impact of global warming [PD03]. Redescription mining allows to study more complex combinations of species than would be otherwise possible with a laborious semi-automatic process requiring the manual selection of species.

Discussion. Notice that the mammals data is Boolean while the climate data is real-valued. Contrarily to the `DBLP` data, the range of the variables is not limited to small natural numbers and the amount of distinct occurring values can be as large as the number of objects. The algorithm determines the optimal discretization on-the-fly and the bounds are fixed to the shortest interval optimizing the accuracy. This can result in interval bounds with many decimals, up to the data precision, making the queries difficult to read. Therefore, taking into account a criterion that favors simple interval bounds could be considered, for the benefit of interpretability.

We note a drift towards redescriptions with largest allowed support cardinality, with a substantial part of the redescriptions covering large areas of the map. As with the bibliographic data, this is again partly an effect of the greater capacity to adjust the queries by tuning the interval bounds. Here, it manifests as fairly complex climatic queries, paired with disjunctions of possibly unrelated species. The resulting redescriptions, while fairly accurate, might be of little interest to biologists. This issue is mitigated by the use of p-values to check the significance of the results, but not adequately resolved yet.

In addition, if the discretization of a given variable at any step of the algorithm is not controlled, very similar candidates could be generated, affecting the diversity of the results.

To summarize, a more complex query language not only increases the search space but also calls for reinforced selection methods. Adding more parameters is not a viable solution. Thus, more holistic approaches, based for example on information theory, need to be explored.

6.4 *Political Candidates Profiles*

With the third illustration we turn to the field of politics.

Dataset. The dataset, dubbed `Elections`, consists of information about the candidates that participated in the 2011 Finnish parliamentary elections. The data was collected from `www.vaalikone.fi`, the "election en-

Table 6.5: Sample of redescriptions from `Elections` mined with ReReMi.

(1)	$q_{\mathbf{L}}$ = party: National Coalition $q_{\mathbf{R}}$ = *Question*: Should authorization be granted for the replacement of the two nuclear reactors at the Loviisa power plant? *Answer*: Yes. ∧ *Question*: Which of the following statements best describes your views regarding Finland's financial support to other euro countries in the crisis? *Answer*: Supporting the euro is in the interest of Finland itself.	J= 0.444	$\|E_{1,1}\|$= 152
(4)	$q_{\mathbf{L}}$ = party: Communist Party $q_{\mathbf{R}}$ = *Question*: What is your opinion on service outsourcing by local authorities to private companies? *Answer*: Outsourced services must be returned to the municipalities. ∧ *Question*: Should Finland apply for NATO membership? *Answer*: Never. ∧ *Question*: What do you think of the current Finnish immigration policy? *Answer*: Too tight.	J= 0.365	$\|E_{1,1}\|$= 57
(5)	$q_{\mathbf{L}}$ = Municipal Representative $q_{\mathbf{R}}$ = *Question*: Recently, Russia banned property ownership by foreigners. On the other hand, Russians have bought thousands of properties in Finland. What should be done? *Answer*: Aquisition should be limited until there is reciprocity.	J= 0.352	$\|E_{1,1}\|$= 441
(7)	$q_{\mathbf{L}}$ = county: Uusimaa $q_{\mathbf{R}}$ = *Question*: State tax revenue is equalized among municipalities, so that the money is transferred from the better-off to poorer municipalities. The largest contributors are Helsinki and Espoo, with approximately 500 million euros transfered to poorer municipalities this year. How should the system react? *Answer*: The metropolitan area should be able to keep a greater percentage of their income tax	J= 0.340	$\|E_{1,1}\|$= 200
(9)	$q_{\mathbf{L}}$ = gender: female $q_{\mathbf{R}}$ = *Question*: In Finland, child benefit is paid for each child until the age of 17, regardless of parental income. Should the sytem be modified? *Importance*: High.	J= 0.321	$\|E_{1,1}\|$= 353
(27)	$q_{\mathbf{L}}$ = county: Lapland $q_{\mathbf{R}}$ = *Question*: Which three countries should Finland befriend first if it were on Facebook? *Answer*: Sweden, Norway and Russia.	J= 0.086	$\|E_{1,1}\|$= 22

gine" of the Finnish newspaper Helsingin Sanomat and made publicly available.[3] One view contains candidate personal profile attributes, such as party, age, and education, while the answers provided to 30 multiple-choice questions and assigned importance form the other view. More precisely, for each of the thirty questions, the candidates were asked to choose the answer that best matched their opinion from a set of suggestions. In addition, they indicated what importance they attach to each question, that is, whether they consider the issue to be of high, medium or low importance. Each attribute-value of the profiles and each distinct question-answer and question-importance pair is represented by a Boolean attribute.

Results. Tables 6.5 and 6.6 present a sample of patterns mined from this dataset with the ReReMi and Translator algorithms respectively. In both cases, the queries consist of monotone conjunctions of Boolean predicates, and we indicate the accuracy and support for each example. In

[3]Data retrieved from `http://blogit.hs.fi/hsnext/hsn-vaalikone-on-nyt-avointa-tietoa` in May 2012.

Table 6.6: Sample of rules from `Elections` mined with TRANSLATOR.

(1)	$q_{\mathbf{L}}$ = party: National Coalition $q_{\mathbf{R}}$ = *Question*: Taxes have increased quickly in Finland since the second half of the 90's. How should this be viewed? *Importance*: Medium. ∧ *Question*: Should authorization be granted for the replacement of the two nuclear reactors at the Loviisa power plant? *Answer*: Yes. ∧ *Question*: Which of the following statements best describes your views regarding Finland's financial support to other euro countries in the crisis? *Answer*: Supporting the euro is in the interest of Finland itself. ∧ *Question*: Which of the following statements most closely matches your vision regarding the global Financial Transaction Tax (FTT) proposed by the EU? *Answer*: The EU should adopt an FTT, even if the rest of the world does not participate in the system. ∧ *Question*: Legistlation regarding arms was tightened in the autumn of 2010, raising the age limit for handgun permits to 20 years. What should be done? *Answer*: The legislation is alright now. ∧ *Question*: What is your opinion on service outsourcing by local authorities to private companies? *Answer*: Outsourcing may be increased, municipalities should learn to improve the quality and prices of their services.	↔	J= 0.211 $\lvert E_{1,1}\rvert$= 48
(11)	$q_{\mathbf{L}}$ = county: Uusimaa $q_{\mathbf{R}}$ = *Question*: State tax revenue is equalized among municipalities, so that the money is transferred from the better-off to poorer municipalities. The largest contributors are Helsinki and Espoo, with approximately 500 million euros transfered to poorer municipalities this year. How should the system react? *Answer*: The metropolitan area should be able to keep a greater percentage of their income tax	←	J= 0.340 $\lvert E_{1,1}\rvert$= 200
(24)	$q_{\mathbf{L}}$ = party: Communist Party $q_{\mathbf{R}}$ = *Question*: Should Finland apply for NATO membership? *Answer*: Never. ∧ *Question*: What do you think of the current Finnish immigration policy? *Answer*: Too tight.	→	J= 0.249 $\lvert E_{1,1}\rvert$= 60
(34)	$q_{\mathbf{L}}$ = party: Social Democratic Party ∧ Municipal Rep. $q_{\mathbf{R}}$ = *Question*: Should Finland apply for NATO membership? *Answer*: Yes, but not at the beginning of the legislature. ∧ *Question*: Recently, Russia banned property ownership by foreigners. On the other hand, Russians have bought thousands of properties in Finland. What should be done? *Answer*: Aquisition should be limited until there is reciprocity.	→	J= 0.111 $\lvert E_{1,1}\rvert$= 53

addition, we indicate the direction of the rules found by the TRANSLATOR algorithm (→, ← or ↔), which are sorted in the order in which they were mined.

In general, the obtained patterns conform to the common understanding of the Finnish political landscape. Redescription 6.5(4), for instance, indicates that the Communist Party of Finland is opposed to the country entering NATO and to the outsourcing of municipal services, while it favors a more permissive immigration policy, opinions commonly attributed to that party.

In many countries, this kind of election recommendation engines, known as Voting Advice Application (VAA), are becoming a common feature at election times [CG10]. Simultaneously, election results, parliamentary activity, or government policies, for example, are made more widely accessible

under the action of open data movements.[4] This offers potential for data analysis tools to promote political awareness among citizens and foster democratic participation. While earning increasing interest and recognition, these initiatives are still in their infancy and present a number of challenges [WNP09, EC13].

The principles underlying redescription mining are simple and interpreting the results requires neither expert training nor extensive domain knowledge. In contrast to more complex analysis methods, which might attract instinctive suspicions of opinion manipulation, this makes the approach suitable for applications targeted at the general public in this field.

Discussion. We observe that the results found by both methods are rather similar, a query pair found by one method often being a subpattern of one found by the other method or vice versa. For instance, redescription 6.5(1) is contained in rule 6.6(1) and vice versa with 6.5(4) and 6.6(24), while 6.5(7) and 6.6(11) have identical queries. However, Translator finds directional rules. For instance, example 6.6(11) indicates that most candidates favorable to Helsinki and Espoo keeping more income tax come from the Uusimaa county, to which both municipalities belong, but that most candidates from that county do not share this opinion. This directional information is absent from ReReMi's results.

The selection of patterns is a major difference between the two algorithms. ReReMi focuses on finding redescriptions with high accuracy while Translator emphasizes the quality of the entire collection of patterns with respect to compression ability. As a result, individual redescriptions found by the first method outmatch those found by the second method with respect to the Jaccard coefficient. The set of translation rules returned by the Translator algorithm includes results that might have a low accuracy or a large p-value. For instance, the p-value of rule 6.6(34) equals 0.11 (see pvalO, in Section 5.1.1) and it would be rejected with most common significance levels. Still, this result set is purportedly more coherent as a whole than the one obtained with ReReMi. In fact, the former allows to compress the data, although in this case the compression ratio is a modest 93%, while the latter actually inflates it with redundancies, with a compression ratio reaching 101%.

As a further advantage, the compression-based method does not require tuning any parameters other than, possibly, a minimum support threshold.

[4] See `http://openelectiondata.org`, `http://www.itsyourparliament.eu` or `http://opengovernmentdata.org`, for example.

However, it is not as scalable as the greedy search and remains to be adapted to more general query languages.

6.5 *Biomedical Ontology*

As the last piece of this exposition, we look at relational redescription mining in the biomedical domain.

Dataset. The `UMLS` dataset, obtained from the Alchemy repository,[5] characterizes the relations between biomedical concepts in terms of the Unified Medical Language System ontology. It can be represented as a network of 135 nodes and 4181 edges. In contrast to the previous examples, this is a relational dataset, containing information about the links between different objects, here biomedical concepts. This particular dataset does not contain information about individual nodes, i.e. there are no node attributes. In this setting, our queries characterize pairs of objects in term of the relations that connect them. The redescriptions we are looking for are pairs of such queries, expressed over disjoint sets of edge attributes, that characterize roughly the same object pairs.

Results. Redescriptions from the `UMLS` dataset mined with the ArRM algorithm are shown in Table 6.7. Alternatively to the graphical representation used in that table, the queries can be written in full textual form, as a conjunction of relational predicates. For instance, redescription 6.7(6) can be written as the following pair of queries:

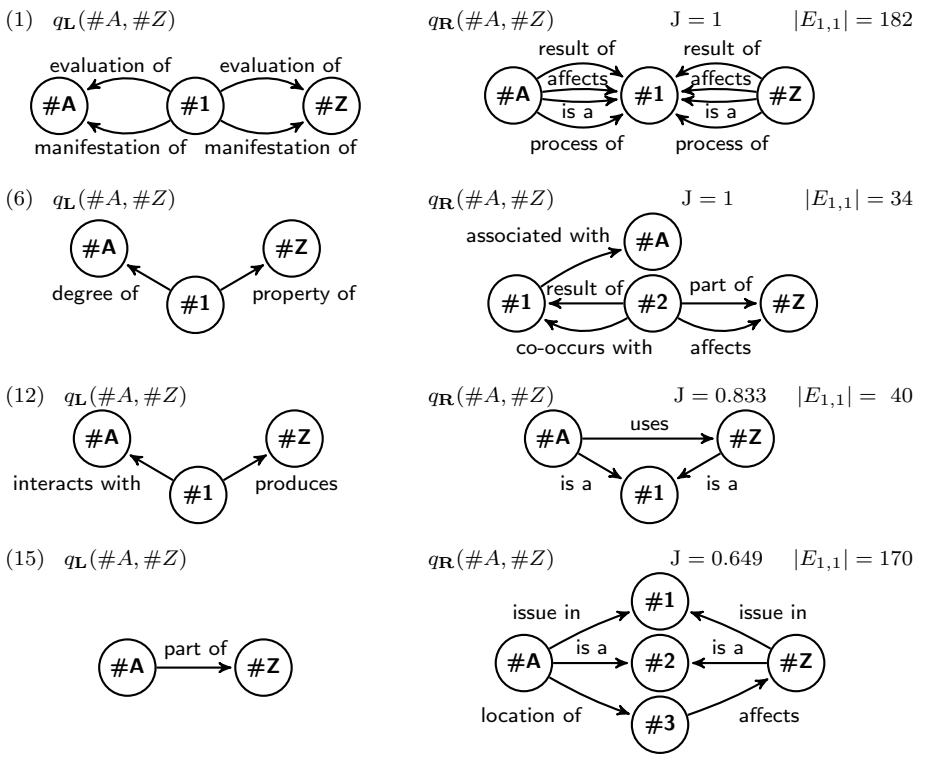
$$\begin{aligned} q_{\mathbf{L}}(\#A, \#Z) \text{ :- } & \epsilon_{\texttt{degree of}}(\#1, \#A), \epsilon_{\texttt{property of}}(\#1, \#Z)\,. \\ q_{\mathbf{R}}(\#A, \#Z) \text{ :- } & \epsilon_{\texttt{associated with}}(\#1, \#A), \epsilon_{\texttt{co-occurs with}}(\#2, \#1), \\ & \epsilon_{\texttt{result of}}(\#2, \#1), \epsilon_{\texttt{part of}}(\#2, \#Z), \epsilon_{\texttt{affects}}(\#2, \#Z)\,. \end{aligned}$$

There are 34 pairs of data objects that map to nodes $\#A$ and $\#Z$ of query $q_{\mathbf{L}}$, and exactly these pairs also can be substituted for nodes $\#A$ and $\#Z$ of query $q_{\mathbf{R}}$. For instance,

$\{\#A$/Organism Attribute, $\#1$/Clinical Attribute, $\#Z$/Amphibian$\}$, and
$\{\#A$/Organism Attribute, $\#1$/Anatomical Abnormality,
$\#2$/Congenital Abnormality, $\#Z$/Amphibian$\}$

[5] Data retrieved from `http://alchemy.cs.washington.edu/data/umls` in Oct. 2012.

Table 6.7: Sample of redescriptions from UMLS mined with Arrm.



are substitutions for query $q_{\mathbf{L}}(\#A, \#Z)$ and $q_{\mathbf{R}}(\#A, \#Z)$, respectively, both corresponding to the object pair (Organism Attribute, Amphibian). Hence, this pair of queries forms a perfect relational redescription, i.e. a redescription of accuracy one, with a support of cardinality 34.

Ontologies are structured formal representations of the concepts within a domain and their relations. Because they define the semantics of the data, ontologies have an important role in the semantic web. In comparison, the schema of a database defines a practical representation of the data in order to allow efficient storage and retrieval, irrespective of meaning. A common problem in order to share information across sources, is to find correspondences between the occurring concepts.

Relational redescription mining provides expressive means to capture nearly equivalent connection patterns between objects in a heterogeneous network. This goes beyond current approaches in ontology alignment [SAS11] and schema matching [SE05] that typically aim to identify one-to-one mappings of concepts or relations.

More generally, relational redescription mining can help explore, understand and maintain complex relational datasets. For instance, it might be useful in large knowledge bases that store millions of objects and relations [ABK+07, CBK+10, SKW07], whose volume makes manual curation impossible.

Discussion. Compared to the propositional setting, while spurious redescriptions are less likely to arise in the relational setting, especially from sparse datasets, finding multiple nearly equivalent redescriptions is a more acute issue. For instance, an added relation or intermediate variable might respectively reduce or increase the number of satisfying substitutions for a query without affecting its support. More simply, a query and subqueries can be satisfied by the same substitutions. For instance, removing relations `is a` and `process of` from the right-hand side query of redescription 6.7(1) does not modify its support. Selecting and filtering such similar patterns, in other words identifying the best representative, is necessary to ensure the quality of the results and requires a tailored solution.

Furthermore, real-world datasets and especially real-world networks are often incomplete and might contain uncertain data, because of the data collection process or the nature of the information. In particular, some data can be inexact and the doubts about the actual values might be modelled as probabilities associated with the data. The problem of handling uncertainties has been considered in relational learning and in other data mining tasks [RKT07, PGdK09, Agg09]. Adapting such approaches to mine redescriptions in the presence of partial and of probabilistic information is thus an important direction for further investigations, in order to increase the practical applicability of the approach.

Chapter 7

Conclusions

The unifying theme of the present thesis is the data analysis task called redescription mining. It aims to find objects that admit multiple shared descriptions and, vice versa, to find distinct common characterizations for a set of objects.

Redescription mining is a task for exploratory data analysis. It provides insight into the data by means of pairs of expressive and interpretable queries, relating different views on the objects. It shares similarities with other data mining tasks like exceptional model mining and subgroup discovery, but is characterized by its symmetrical approach.

In this thesis, we extended redescription mining beyond propositional Boolean queries to real-valued attributes and relational queries. We designed the ReReMi algorithm to mine redescriptions over nominal and real-valued attributes natively and introduced the Arrm relational redescription mining algorithm.

We also proposed two approaches for selecting high quality redescriptions. The Siren interface for mining and visualizing redescriptions, on one hand, enables the user to interactively adjust the selection criteria. The Translator algorithm, on the other hand, provides a principled solution to the selection problem. It is a parameter-free compression-based algorithm that encodes one side of the data using the other side, and vice versa, thereby capturing the associations across the two sides.

While its underlying principle is simple and intuitive, we showed that redescription mining constitutes a powerful tool for data exploration, potentially applicable in a large variety of domains.

Further developing the algorithms presented here and integrating them together should help alleviate current shortcomings such as spurious or redundant results and the absence of any analytical guarantee on finding the best redescriptions occurring in the data. The scalability of the algorithms and their generalization to varying numbers of views also demand investigation.

Specifically, devising methods with sound theoretic foundations and sufficient flexibility to select redescriptions, for instance drawing on recent advances in significance testing for data mining [Oja11, Han12, Vuo12] or modelling the information content of redescriptions in the subjective interestingness framework [DB11a], constitutes a major direction for future research.

Besides, uncertainties are inherent to most real-world scenarios. To promote its applicability in realistic situations, redescription mining should thus be enabled to account for uncertainties in the data, possibly by adapting techniques developed for other data analysis tasks [Agg09].

Finally, the actual value of our proposed methods can only be assessed by putting them to use, in collaboration with experts and practitioners of the respective fields.

References

[ABK+07] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. G. DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference and the 2nd Asian Semantic Web Conference, ISWC/ASWC'07* (Busan, Korea), volume 4825 of *Lecture Notes in Computer Science*, pages 722–735. Springer, 2007.

[Agg09] Aggarwal, C. C. *Managing and Mining Uncertain Data*, volume 35 of *Advances in Database Systems*. Springer, 2009.

[AIS93] Agrawal, R., Imielinski, T., and Swami, A. N. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, SIGMOD'93* (Washington, D.C., U.S.A.), pages 207–216. ACM Press, 1993.

[AS94] Agrawal, R., and Srikant, R. Fast algorithms for mining association rules in large databases. In *Proceedings of 20th International Conference on Very Large Data Bases, VLDB'94* (Santiago de Chile, Chile), pages 487–499. Morgan Kaufmann, 1994.

[BB00] Boulicaut, J.-F., and Bykowski, A. Frequent closures as a concise representation for binary data mining. In *Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications, PADKK'00* (Kyoto, Japan), volume 1805 of *Lecture Notes in Computer Science*, pages 62–73. Springer, 2000.

[BDRM05] Boulicaut, J.-F., De Raedt, L., and Mannila, H., editors. *Revised Selected Papers of the European Workshop on Inductive Databases and Constraint Based Mining*, volume 3848 of *Lecture Notes in Computer Science*. Springer, 2005.

[BKM99] Boulicaut, J.-F., Klemettinen, M., and Mannila, H. Modeling KDD processes within the inductive database framework. In *Proceedings of the 1st International Conference on Data Warehousing and Knowledge Discovery, DaWaK'99* (Florence, Italy), volume 1676 of *Lecture Notes in Computer Science*, pages 293–302. Springer, 1999.

[BM98] Blum, A., and Mitchell, T. M. Combining labeled and unlabeled sata with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory, COLT'98* (Madison, Wisconsin, U.S.A.), pages 92–100. ACM, 1998.

[BMS97] Brin, S., Motwani, R., and Silverstein, C. Beyond market baskets: Generalizing association rules to correlations. In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, SIGMOD'97* (Tucson, Arizona, U.S.A.), pages 265–276. ACM Press, 1997.

[BS04] Bickel, S., and Scheffer, T. Multi-view clustering. In *Proceedings of the 4th IEEE International Conference on Data Mining, ICDM'04* (Brighton, UK), pages 19–26. IEEE Computer Society, 2004.

[BS05] Bickel, S., and Scheffer, T. Estimation of mixture models using co-EM. In *Proceedings of the 16th European Conference on Machine Learning, ECML'05* (Porto, Portugal), volume 3720 of *Lecture Notes in Computer Science*, pages 35–46. Springer, 2005.

[CBK+10] Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, Jr., E. R., and Mitchell, T. M. Toward an architecture for never-ending language learning. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence, AAAI'10* (Atlanta, Georgia, U.S.A.). AAAI Press, 2010.

[CG02] Calders, T., and Goethals, B. Mining all non-derivable frequent itemsets. In *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery, PKDD'02* (Helsinki, Finland), volume 2431 of *Lecture Notes in Computer Science*, pages 74–85. Springer, 2002.

[CG10] Cedroni, L., and Garzia, D., editors. *Voting Advice Applications in Europe: The State of the Art*. ScriptaWeb, Naples, 2010.

[Cod70] Codd, E. F. A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387, 1970.

[CRB04] Calders, T., Rigotti, C., and Boulicaut, J.-F. A survey on condensed representations for frequent sets. In Boulicaut et al. [BDRM05], pages 64–80.

[DB11a] De Bie, T. An information theoretic framework for data mining. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'11* (San Diego, California, U.S.A.), pages 564–572. ACM, 2011.

[DB11b] De Bie, T. Maximum entropy models and subjective interestingness: An application to tiles in binary databases. *Data Mining and Knowledge Discovery*, 23(3):407–446, 2011.

[DEDC96] Deransart, P., Ed-Dbali, A., and Cervoni, L. *Prolog: the Standard: Reference Manual*. Springer Verlag, 1996.

[DEV12] Dinh, Q.-T., Exbrayat, M., and Vrain, C. A link-based method for propositionalization. In *Late Breaking Papers of the 22nd International Conference on Inductive Logic Programming, ILP'12* (Dubrovnik, Croatia), volume 975 of *CEUR Workshop Proceedings*, pages 10–25, 2012.

[DR02] De Raedt, L. A perspective on inductive databases. *SIGKDD Explorations*, 4(2):69–77, 2002.

[DR08] De Raedt, L. *Logical and Relational Learning*. Springer, 2008.

[DRGN08] De Raedt, L., Guns, T., and Nijssen, S. Constraint programming for itemset mining. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'08* (Las Vegas, Nevada, U.S.A.), pages 204–212. ACM, 2008.

[DRR04] De Raedt, L., and Ramon, J. Condensed representations for inductive logic programming. In *Proceedings of the Ninth International Conference on Principles of Knowledge Representation and Reasoning, KR'04* (Whistler, British Columbia, Canada), pages 438–446. AAAI Press, 2004.

[DT99] Dehaspe, L., and Toivonen, H. Discovery of frequent DATALOG patterns. *Data Mining and Knowledge Discovery*, 3(1):7–36, 1999.

[EC13] Evans, A. M., and Campos, A. Open government initiatives: Challenges of citizen participation. *Journal of Policy Analysis and Management*, 32(1):172–185, 2013.

[Edg95] Edgington, E. S. *Randomization Tests (3rd Edition)*, volume 147. CRC Press, 1995.

[EMS+94] Esposito, F., Malerba, D., Semeraro, G., Brunk, C., and Pazzani, M. Traps and pitfalls when learning logical definitions from relations. In *Proceedings of the 8th International Symposium on Methodologies for Intelligent Systems, ISMIS'94* (Charlotte, North Carolina, U.S.A.), volume 869 of *LNCS*, pages 376–385. Springer, 1994.

[FHM+05] Farquhar, J. D. R., Hardoon, D. R., Meng, H., Shawe-Taylor, J., and Szedmák, S. Two view learning: SVM-2K, theory and practice. In *Proceedings of the 19th Annual Conference on Neural Information Processing Systems, NIPS'05* (Vancouver, British Columbia, Canada), 2005.

[Fis38] Fisher, R. *Statistical Methods for Research Workers*. Biological monographs and manuals. Oliver and Boyd, 1938.

[GDJE12] Günnemann, S., Dao, P., Jamali, M., and Ester, M. Assessing the significance of data mining results on graphs with feature vectors. In *Proceedings of the 12th IEEE International Conference on Data Mining, ICDM'12* (Brussels, Belgium), pages 270–279. IEEE Computer Society, 2012.

[GMM08] Gallo, A., Miettinen, P., and Mannila, H. Finding subgroups having several descriptions: Algorithms for redescription mining. In *Proceedings of the 8th SIAM International Conference on Data Mining, SDM'08* (Atlanta, Georgia, U.S.A.), pages 334–345. SIAM, 2008.

[GMMT07] Gionis, A., Mannila, H., Mielikäinen, T., and Tsaparas, P. Assessing data mining results via swap randomization. *ACM Transactions on Knowledge Discovery from Data*, 1(3), 2007.

[GNDR13] Guns, T., Nijssen, S., and De Raedt, L. k-Pattern set mining under constraints. *IEEE Transactions on Knowledge and Data Engineering*, 25(2):402–418, 2013.

[GNZDR11] Guns, T., Nijssen, S., Zimmermann, A., and De Raedt, L. Declarative heuristic search for pattern set mining. In *Proceedings of the 2011 IEEE International Conference on Data Mining Workshops, ICDMW'11* (Vancouver, British Columbia, Canada), pages 1104–1111. IEEE Computer Society, 2011.

[Goe03] Goethals, B. Survey on frequent pattern mining. Manuscript, 2003.

[GR00] Garofalakis, M. N., and Rastogi, R. Scalable data mining with model constraints. *SIGKDD Explorations*, 2(2):39–48, 2000.

[Gri17] Grinnell, J. The niche-relationships of the California Thrasher. *The Auk*, 34(4):427–433, 1917.

[Grü07] Grünwald, P. D. *The Minimum Description Length Principle*. Adaptative computation and machine learning series. MIT Press, 2007.

[Han12] Hanhijärvi, S. *Multiple Hypothesis Testing in Data Mining*. PhD thesis, Aalto University School of Science, Department of Information and Computer Science, Finland, 2012.

[HCP+05] Hijmans, R. J., Cameron, S., Parra, L., Jones, P., and Jarvis, A. Very high resolution interpolated climate surfaces for global land areas. *International Journal of Climatology*, 25:1965–1978, 2005. `www.worldclim.org`.

[HCXY07] Han, J., Cheng, H., Xin, D., and Yan, X. Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 15(1):55–86, 2007.

[HGP09] Hanhijärvi, S., Garriga, G. C., and Puolamäki, K. Randomization techniques for graphs. In *Proceedings of the 2009 SIAM International Conference on Data Mining, SDM'09* (Sparks, Nevada, U.S.A.), pages 780–791. SIAM, 2009.

[HK00] Han, J., and Kamber, M. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2000.

[HMS01] Hand, D., Mannila, H., and Smyth, P. *Principles of Data Mining*. Adaptative computation and machine learning series. MIT Press, 2001.

[HPYM04] Han, J., Pei, J., Yin, Y., and Mao, R. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Mining and Knowledge Discovery*, 8(1):53–87, 2004.

[IM96] Imielinski, T., and Mannila, H. A database perspective on knowledge discovery. *Communications of the ACM*, 39(11):58–64, 1996.

[JMR08] Jin, Y., Murali, T. M., and Ramakrishnan, N. Compositional mining of multirelational biological datasets. *ACM Transactions on Knowledge Discovery from Data*, 2(1), 2008.

[KBC10] Khiari, M., Boizumault, P., and Crémilleux, B. Constraint programming for mining n-ary patterns. In *Proceedings of the 16th International Conference on Principles and Practice of Constraint Programming, CP'10* (St. Andrews, Scotland, UK), volume 6308 of *Lecture Notes in Computer Science*, pages 552–567. Springer, 2010.

[KK08] Klami, A., and Kaski, S. Probabilistic approach to detecting dependencies between data sets. *Neurocomputing*, 72(1-3):39–46, 2008.

[KMR+94] Klemettinen, M., Mannila, H., Ronkainen, P., Toivonen, H., and Verkamo, A. I. Finding interesting rules from large sets of discovered association rules. In *Proceedings of the 1994 ACM International Conference on Information and Knowledge Management, CIKM'94* (Gaithersburg, Maryland, U.S.A.), pages 401–407. ACM, 1994.

[Kum07] Kumar, D. *Redescription Mining: Algorithms and Applications in Bioinformatics*. PhD thesis, Department of Computer Science, Virginia Tech, U.S.A., 2007.

[KZ09] Kuzelka, O., and Zelezný, F. Block-wise construction of acyclic relational features with monotone irreducibility and relevancy properties. In *Proceedings of the 26th Annual International*

Conference on Machine Learning, ICML'09 (Montreal, Quebec, Canada), volume 382 of *ACM International Conference Proceeding Series*, pages 569–576. ACM, 2009.

[LFK08] Leman, D., Feelders, A., and Knobbe, A. J. Exceptional model mining. In *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases, ECML/PKDD'08, Part II* (Antwerp, Belgium), volume 5212 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2008.

[LR05] Lehmann, E. E. L., and Romano, J. P. *Testing Statistical Hypotheses*. Springer Science+ Business Media, 2005.

[Man00] Mannila, H. Theoretical frameworks for data mining. *SIGKDD Explorations*, 1(2):30–32, 2000.

[MBKN13] Martin, T., Ball, B., Karrer, B., and Newman, M. E. J. Coauthorship and citation in scientific publishing. *arXiv preprint arXiv:1304.0473*, 2013.

[MDR94] Muggleton, S., and De Raedt, L. Inductive logic programming: Theory and methods. *Journal of Logic Programming*, 19/20:629–679, 1994.

[MJAB+99] Mitchell-Jones, A. J., Amori, G., Bogdanowicz, W., Krystufek, B., Reijnders, P., Spitzenberger, F., Stubbe, M., Thissen, J., Vohralik, V., and Zima, J. *The Atlas of European Mammals*. Academic Press, London, 1999. `www.european-mammals.org`.

[MS98] Megiddo, N., and Srikant, R. Discovering predictive association rules. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, KDD'98* (New York City, New York, U.S.A.), pages 274–278. AAAI Press, 1998.

[MT96] Mannila, H., and Toivonen, H. Multiple uses of frequent sets and condensed representations (extended abstract). In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD'96* (Portland, Oregon, U.S.A.), pages 189–194. AAAI Press, 1996.

[MT97] Mannila, H., and Toivonen, H. Levelwise search and borders of theories in knowledge discovery. *Data Mining and Knowledge Discovery*, 1(3):241–258, 1997.

[MTV94] Mannila, H., Toivonen, H., and Verkamo, A. I. Efficient algorithms for discovering association rules. In *Proceedings of the 1994 AAAI Workshop on Knowledge Discovery in Databases, KDD'94, Technical Report WS-94-03* (Seattle, Washington, U.S.A.), pages 181–192. AAAI Press, 1994.

[Mug95] Muggleton, S. Inverse entailment and progol. *New Generation Computing*, 13(3&4):245–286, 1995.

[New01a] Newman, M. E. J. Scientific collaboration networks. i. network construction and fundamental results. *Physical Review E*, 64:016131, 2001.

[New01b] Newman, M. E. J. Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. *Physical Review E*, 64:016132, 2001.

[NG00] Nigam, K., and Ghani, R. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the 2000 ACM International Conference on Information and Knowledge Management, CIKM'00* (McLean, Virginia, U.S.A.), pages 86–93. ACM, 2000.

[NLHP98] Ng, R. T., Lakshmanan, L. V. S., Han, J., and Pang, A. Exploratory mining and pruning optimizations of constrained association rules. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, SIGMOD'98* (Seattle, Washington, U.S.A.), pages 13–24. ACM Press, 1998.

[NLW09] Novak, P. K., Lavrac, N., and Webb, G. I. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *Journal of Machine Learning Research*, 10:377–403, 2009.

[Oja11] Ojala, M. *Randomization Algorithms for Assessing the Significance of Data Mining Results.* PhD thesis, Aalto University School of Science, Department of Information and Computer Science, Finland, 2011.

[PBTL99] Pasquier, N., Bastide, Y., Taouil, R., and Lakhal, L. Discovering frequent closed itemsets for association rules. In *Proceedings of the 7th International Conference on Database Theory, ICDT'99* (Jerusalem, Israel), volume 1540 of *Lecture Notes in Computer Science*, pages 398–416. Springer, 1999.

[PCY95] Park, J. S., Chen, M.-S., and Yu, P. S. An effective hash based algorithm for mining association rules. In *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data, SIGMOD'95* (San Jose, California, U.S.A.), pages 175–186. ACM Press, 1995.

[PD03] Pearson, R. G., and Dawson, T. P. Predicting the impacts of climate change on the distribution of species: Are bioclimate envelope models useful? *Global Ecology and Biogeography*, 12:361–371, 2003.

[PGdK09] Pei, J., Getoor, L., and de Keijzer, A., editors. *Proceedings of the 1st ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data, U'09.* ACM, 2009.

[PR05] Parida, L., and Ramakrishnan, N. Redescription mining: Structure theory and algorithms. In *Proceedings of the 20th National Conference on Artificial Intelligence and the 7th Innovative Applications of Artificial Intelligence Conference, AAAI'05* (Pittsburgh, Pennsylvania, U.S.A.), pages 837–844. AAAI Press / The MIT Press, 2005.

[PSM94] Piatetsky-Shapiro, G., and Matheus, C. J. The interestingness of deviations. In *Proceedings of the 1994 AAAI Workshop on Knowledge Discovery in Databases, KDD'94, Technical Report WS-94-03* (Seattle, Washington, U.S.A.), pages 25–36. AAAI Press, 1994.

[PT98] Padmanabhan, B., and Tuzhilin, A. A belief-driven method for discovering unexpected patterns. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, KDD'98* (New York City, New York, U.S.A.), pages 94–100. AAAI Press, 1998.

[PT00] Padmanabhan, B., and Tuzhilin, A. Small is beautiful: Discovering the minimal set of unexpected patterns. In *Proceedings of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD'00* (Boston, Massachusetts, U.S.A.), pages 54–63. ACM, 2000.

[QCJ93] Quinlan, J. R., and Cameron-Jones, R. M. FOIL: A midterm report. In *Proceedings of the 4th European Conference on Machine Learning, ECML'93* (Vienna, Austria), volume 667

of *Lecture Notes in Computer Science*, pages 3–20. Springer, 1993.

[RKM+04] Ramakrishnan, N., Kumar, D., Mishra, B., Potts, M., and Helm, R. F. Turning CARTwheels: An alternating algorithm for mining redescriptions. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'04* (Seattle, Washington, U.S.A.), pages 266–275. ACM, 2004.

[RKT07] Raedt, L. D., Kimmig, A., and Toivonen, H. ProbLog: A probabilistic prolog and its application in link discovery. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI'07* (Hyderabad, India), pages 2462–2467, 2007.

[RLT+12] Rettinger, A., Lösch, U., Tresp, V., d'Amato, C., and Fanizzi, N. Mining the semantic web — statistical learning for next generation knowledge bases. *Data Mining and Knowledge Discovery*, 24(3):613–662, 2012.

[SAS11] Suchanek, F. M., Abiteboul, S., and Senellart, P. PARIS: Probabilistic alignment of relations, instances, and schema. *Proceedings of the VLDB Endowment (PVLDB)*, 5(3):157–168, 2011.

[SE05] Shvaiko, P., and Euzenat, J. A survey of schema-based matching approaches. *Journal on Data Semantics IV*, pages 146–171, 2005.

[SKW07] Suchanek, F. M., Kasneci, G., and Weikum, G. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW'07* (Banff, Alberta, Canada), pages 697–706. ACM, 2007.

[SN09] Soberón, J., and Nakamura, M. Niches and distributional areas: Concepts, methods, and assumptions. *Proceedings of the National Academy of Sciences of the United States, PNAS*, 106(Supplement 2):19644–19650, 2009.

[Sri07] Srinivasan, A. *The Aleph Manual*. University of Oxford, 2007.

[ST95] Silberschatz, A., and Tuzhilin, A. On subjective measures of interestingness in knowledge discovery. In *Proceedings of the*

First International Conference on Knowledge Discovery and Data Mining, KDD'95 (Montreal, Quebec, Canada), pages 275–281. AAAI Press, 1995.

[SVA97] Srikant, R., Vu, Q., and Agrawal, R. Mining association rules with item constraints. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, KDD'97* (Newport Beach, California, U.S.A.), pages 67–73. AAAI Press, 1997.

[TCP09] Tran, Q. T., Chan, C.-Y., and Parthasarathy, S. Query by output. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data, SIGMOD'09* (Providence, Rhode Island, U.S.A.), pages 535–548. ACM Press, 2009.

[TKOK11] Tripathi, A., Klami, A., Oresic, M., and Kaski, S. Matching samples of multiple views. *Data Mining and Knowledge Discovery*, 23(2):300–321, 2011.

[TPB00] Tishby, N., Pereira, F., and Bialek, W. The information bottleneck method. *arXiv preprint arXiv:physics/0004057*, 2000.

[UZT+09] Umek, L., Zupan, B., Toplak, M., Morin, A., Chauchat, J.-H., Makovec, G., and Smrke, D. Subgroup discovery in data sets with multi-dimensional responses: A method and a case study in traumatology. In *Proceedings of the 12th Conference on Artificial Intelligence in Medicine, AIME'09* (Verona, Italy), volume 5651 of *Lecture Notes in Computer Science*, pages 265–274, 2009.

[VKKK12] Virtanen, S., Klami, A., Khan, S. A., and Kaski, S. Bayesian group factor analysis. *Journal of Machine Learning Research - Proceedings Track*, 22:1269–1277, 2012.

[Vuo12] Vuokko, N. *Testing the Significance of Patterns with Complex Null Hypotheses*. PhD thesis, Aalto University School of Science, Department of Information and Computer Science, Finland, 2012.

[VvLS11] Vreeken, J., van Leeuwen, M., and Siebes, A. Krimp: Mining itemsets that compress. *Data Mining and Knowledge Discovery*, 23(1):169–214, 2011.

[Web07] Webb, G. I. Discovering significant patterns. *Machine Learning*, 68(1):1–33, 2007.

[WNP09] Walgrave, S., Nuytemans, M., and Pepermans, K. Voting aid applications and the effect of statement selection. *West European Politics*, 32(6):1161–1180, 2009.

[Yar95] Yarowsky, D. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, ACL'95* (MIT, Cambridge, Massachusetts, U.S.A.), pages 189–196. Morgan Kaufmann Publishers / ACL, 1995.

[ZH02] Zaki, M. J., and Hsiao, C.-J. CHARM: An efficient algorithm for closed itemset mining. In *Proceedings of the Second SIAM International Conference on Data Mining, SDM'02* (Arlington, Virginia, U.S.A.). SIAM, 2002.

[ZPT04] Zhang, H., Padmanabhan, B., and Tuzhilin, A. On the discovery of significant statistical quantitative rules. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'04* (Seattle, Washington, U.S.A.), pages 374–383. ACM, 2004.

[ZR05] Zaki, M. J., and Ramakrishnan, N. Reasoning about sets using redescription mining. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'05* (Chicago, Illinois, U.S.A.), pages 364–373. ACM, 2005.