Preface

This ECML/PKDD 2005 workshop, follow-up to the successful KDO-04 Workshop held at the ECML/PKDD 2004 in Pisa as well as the (semantic) web mining workshops at ECML/PKDD 2001-2003, was concerned with the interaction between prior knowledge as encoded in *ontologies* and derived knowledge as obtained by a *knowledge discovery* process.

Early approaches to KDD typically relied on one-size-fits-all solutions. More recently, however, the role of available prior knowledge as well as specific profile of the user have been increasingly taken into account. Such contextual information may help select the suitable data, prune the space of hypothesis and represent the output in a most comprehensible way. Ontological grounding is a pre-requisite for efficient automated usage of such information with respect to a particular mining session. Notably, availability of domain ontologies also enables to automatically expose the mining results on the semantic web, to provide some KDD tools in the form of (semantic) web services, or to handle heterogeneous and complex objects when mining web data for the purpose of (semantic) web personalisation.

In some domains large bodies of consensual knowledge already exist. This is the case in medicine: although e.g. UMLS or Foundational Model of Anatomy are not ideal ontologies (i.e. formal theories) in the strictly logical sense, they express large-scale and long-term pragmatic structuring of domain knowledge. In many other domains, however, it might be necessary to start from a collection of data (esp. text) and to design the first version of ontology inductively, via ontology learning. The result of ontology learning (possibly fine-tuned by the human designer) eventually becomes input for further empirical discovery.

While knowledge engineering research has already recognized the value of knowledge discovery from textual and semi-structured resources in the process of building an ontology (i.e. in ontology learning), links in the opposite direction are more rare. Within the context of this workshop we intended to bring together researchers from both directions in order to initiate a discussion on how to integrate insights from both communities.

We obtained 15 paper submissions; each of them has been evaluated by three reviewers. Based on the reviewers' opinions, we selected 9 submissions for presentation. All papers were presented orally at the workshop. The texts of all accepted contributions are collected in this volume.

Although the papers shared the unifying idea of using KDD for ontology development and/or vice versa, the underlying KDD techniques, forms of data (e.g. text/tabular) as well as ontology types—never mind application domains—were fairly heterogeneous.

The papers could more or less be divided into two almost well balanced groups. Namely, into a *'KDD for Ontology'* segment, which deals with ontology generation, evaluation and manipulation:

– d'Amato et. al. define a new dissimilarity measure for ontological concept descriptions in $\mathcal{ALC}$,
– Dupret and Piwowarski show how a singular value decomposition of a term similarity matrix induces a term taxonomy,
– Fortuna et. al. present a system for semi-automatic topic ontology construction that integrates two well-known methods, latent semantic indexing and k-means clustering, for topic discovery,
– Lendvai makes use of the document structure of semantically annotated medical documents to extract a conceptual taxonomy,
– Spiliopoulou et. al. present a method for the evaluation of ontology establishment and enhancement tools by combining perceived quality and statistical measurements.

And an *'Ontology for KDD'* segment, which presents contributions dealing with the application of ontologies in KDD tasks and processes:

– Bogorny et. al. present an approach to reduce the number of spatial relationships for knowledge discovery in geographical databases, using a geoontology and semantic spatial integrity constraints,
– Domingues and Rezende present an algorithm and a system for the generalization and analysis of association rules by the use of taxonomies,
– Litvak et. al. show how the classification of multi-lingual web documents can be improved by the usage of a domain ontology,
– Svátek et. al., finally, used a domain ontology to find explanations for discovered associations in social modelling.

The workshop also featured an excellent invited talk by Magdalina Eirinaki and Michalis Vazirgiannis, which brought together knowledge discovery and ontologies to solve a third problem, namely that of Web personalization.

The organizers would like to thank all presenters for inspiring talks and presentations, the PC members and additional reviewers for their careful work, the local organizers from the ECML/PKDD 2005 team, and last but not least the Workshop Chair.

September 2005

Markus Ackermann
Bettina Berendt
Marko Grobelnik
Vojtěch Svátek
Workshop Chairs
KDO'05

Organization

Program Chairs and Organizing Committee

- Markus Ackermann, University of Leipzig, Germany
- Bettina Berendt, Humboldt University, Berlin, Germany
- Marko Grobelnik, Jozef Stefan Inst., Ljubljana, Slovenia
- Vojtěch Svátek, Univ. of Economics, Prague, Czech Republic

Program Committee

- Nathalie Aussenac-Gilles, IRIT, Toulouse, France
- Abraham Bernstein, Univ. of Zürich, Switzerland
- Christian Biemann, University of Leipzig, Germany
- Paul Buitelaar, DFKI, Saarbrücken, Germany
- Mario Cannataro, Univ. of Catanzaro, Italy
- Philipp Cimiano, AIFB, University of Karlsruhe, Germany
- Martine Collard, Univ. of Nice, France
- Aldo Gangemi, ISTC Roma, Italy
- Andreas Hotho, University of Kassel, Germany
- Mike Jackson, University of Central England, UK
- Francois Jacquenet, University of Saint-Etienne, France
- Alipio Jorge, University of Porto, Portugal
- Nada Lavrac, Jozef Stefan Institute, Slovenia
- Bing Liu, University of Illinois, USA
- Bernardo Magnini, ITC-IRST, Trento, Italy
- Dunja Mladenic, Jozef Stefan Institute, Ljubljana
- Bamshad Mobasher, DePaul University, USA
- Gerhard Paaß, Fraunhofer AIS, St. Augustin, Germany
- Georgios Paliouras, NCSR *Demokritos*, Athens, Greece
- John Punin, Oracle Corporation, USA
- Jan Rauch, University of Economics, Prague, Czech Republic
- Massimo Ruffolo, ICAR-CNR & EXEURA, Italy
- Michael Sintek, DFKI, Kaiserslautern, Germany
- Derek Sleeman, University of Aberdeen, UK
- Steffen Staab, Univ. of Koblenz, Germany
- Gerd Stumme, Univ. of Kassel, Germany
- York Sure, Univ. of Karlsruhe, Germany
- Domenico Talia, University of Calabria, Italy
- Stefan Wrobel, Fraunhofer AIS, St. Augustin, Germany

Table of Contents

Ontologies in Web Personalisation (Invited Talk)

Magdalini Eirinaki and Michalis Vazirgiannis

Athens University of Economics & Business, Department of Informatics
Athens, Greece
{eirinaki, mvazirg}@aueb.gr

Abstract

Web personalization is the process of customizing a web site to the needs of each specific user or set of users. Personalization of a web site may be performed by the provision of recommendations to the users, highlighting/adding links, creation of index pages, etc. Most of the research efforts in Web personalization correspond to the evolution of extensive research in Web usage mining, i.e. the exploitation of the navigational patterns of the web sites visitors. When a personalization system relies solely on usage-based results, however, valuable information conceptually related to what is finally recommended may be missed. The exploitation of the web pages semantics can considerably improve the results of web usage mining and personalization, since it provides a more abstract yet uniform and both machine and human understandable way of processing and analyzing the usage data. The underlying idea is to integrate usage data with content semantics, expressed in ontology terms, in order to produce semantically enhanced navigational patterns that can subsequently be used for producing valuable recommendations. In this talk we will present how a personalization system may benefit from incorporating semantics in the web usage mining and personalization process.

Invited Speakers' Short Biographies

Magdalini Eirinaki is currently a senior PhD candidate in the Department of Informatics of Athens University of Economics and Business (since 2002). She holds a degree in Informatics from the University of Piraeus (1998), and an MSc degree (with Distinction) in Advanced Computing from the Imperial College of Science, Technology and Medicine, University of London (2000). Her interests cover the Web mining and Web personalization areas. Her research work focuses on the integration of Web usage and content semantics, as well as the involvement of link analysis techniques in the Web personalization process. She has served as a reviewer in international conferences and journals and has published several papers in international refereed journals, conferences and workshops. She has also contributed a chapter in an international book.

Michalis Vazirgiannis is an Associate Professor in the Department of Informatics of Athens Univ. of Economics & Business. He holds a degree in Physics (1986), a MSc. in Robotics (1988), and a MSc. In Knowledge Based Systems. In 1994 he obtained a Ph.D. degree in Informatics. Since then, he has conducted research in the Knowledge & DB Lab (of N.T.U. Athens, Greece), in GMD–IPSI (Darmstadt, Germany), in Fern-Universitaet (Hagen, Germany), in project VERSO in INRIA/Paris, in IBM India Research Laboratory, and in Max Planck Institut fü Informatik (Germany). His research interests and work range from data mining to web content management, spatiotemporal databases, and global computing. He has been awarded the ERCIM post doctoral fellowship in the year 2001. In February 2005 he received the MARIE CURIE intra European fellowship to visit INRIA/Paris. The fellowship supports research in the area of P2P web indexing and searching. He is currently the scientific person in charge of two EU funded competitive research programs and several national ones. He served as a conference committee member and as reviewer for international conferences and journals. He has published two books with Springer-Verlag, contributed chapters in international books and encyclopaedias and more than sixty articles in international journals and conference proceedings.

A Semantic Dissimilarity Measure for Concept Descriptions in Ontological Knowledge Bases

Claudia d'Amato, Nicola Fanizzi, Floriana Esposito

Dipartimento di Informatica, Università degli Studi di Bari
Campus Universitario, Via Orabona 4, 70125 Bari, Italy
{claudia.damato,fanizzi,esposito}@di.uniba.it

Abstract. This work presents a dissimilarity measure for expressive Description Logics that are the theoretical counterpart of the standard representations for ontological knowledge. The focus is on the definition of a dissimilarity measure for the $\mathcal{ALC}$ description logic based both on the syntax and on the semantics of the descriptions.

1 Introduction

Recent investigations have emphasized the use of ontologies similarity measures for *Information Retrieval* and *Integration* [1, 2]. However, there is a number of other tasks that may exploit similarity measures, such as, for instance, clustering by means of *partitional* or *agglomerative* algorithms. Therefore, in a Semantic Web perspective, similarity measures can enable such algorithms to exploit the available ontological knowledge expressed in suitable representations, namely concept languages which are candidate as standard in this context.

Various measures for concept representations have been proposed in the literature. A measure has been proposed as a function of the *path distance* between terms in the hierarchical structure underlying the ontology [3]. Other methods for assessing the similarity of concept descriptions are based on *feature matching* [4] and *information content* [5]. The former approach uses both common and discriminant features among concepts and/or concept instances to compute the semantic similarity. The latter method is founded on *Information Theory*. A similarity measure for concepts within a hierarchy is defined in terms of the amount of information conveyed by their immediate super-concept. This is a measure of the variation of information from a description level to a more general one.

Other measures compute the similarity among classes (concepts) belonging to different ontologies. In [6] a number of measures is presented for comparing concepts located in possibly heterogeneous ontologies. The following requirements are made: the formal representation supports inferences such as *subsumption* and local concepts in different ontologies inherit their definitional structure from concepts in a shared ontology. In particular, the intersection of the sets of concept instances is assumed as an indication of the correspondence between these concepts. In [7] a similarity function determines similar classes by using a matching process making use of synonym sets, semantic neighborhood, and discriminating features that are classified into parts, functions, and attributes.

Most of the cited works adopt a semantic approach in conjunction with the structure of the considered descriptions. Besides, the syntactic structure of the descriptions becomes much less important when richer representations are adopted due to the expressive operators that can be employed.

Most of these works focussed on the similarity of atomic concepts (within a hierarchy) rather than on composite ones. Nevertheless, the standard ontology markup languages (e.g., OWL) are founded in Description Logics (DLs) since they borrow the typical DLs constructors. Thus, it becomes necessary to investigate the similarity of complex concept descriptions expressed in DLs. In this respect, to the best of our knowledge, there has been no comparable effort in the literature, except the ideas in [8].

In this position paper, we introduce a semantic dissimilarity measure between descriptions which is suitable for an expressive DL like $\mathcal{ALC}$ [9]. The measure is based on the underlying semantics elicited by querying the knowledge base, as proposed also in [10]. Moreover, recurring the notion of *most specific concept* of an individual, the measure is extended to the individual-concept and individual-individual cases, which may be exploited in knowledge discovery settings.

2 The Reference Representation Language

The basics of $\mathcal{ALC}$ [9] are recalled, a logic which is sufficiently expressive to support most of the constructors of standard ontology languages.

Primitive *concepts*, denoted with names from $N_C = \{C, D, \ldots\}$, are interpreted as subsets of a domain of objects and primitive *roles*, denoted with names taken from $N_R = \{R, S, \ldots\}$, are interpreted as binary relations on such a domain. Complex descriptions are built using primitive concepts and roles and the constructors in Table 1. The meaning is defined by an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the *domain* of the interpretation and $\cdot^{\mathcal{I}}$ is the *interpretation function*, mapping the intension of concepts and roles to their extension.

A *knowledge base* $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ contains a *T-box* $\mathcal{T}$ and an *A-box* $\mathcal{A}$. $\mathcal{T}$ is a set of definitions $C \equiv D$, meaning $C^{\mathcal{I}} = D^{\mathcal{I}}$, where C is the concept name and D is a description as defined above. $\mathcal{A}$ contains extensional assertions on concepts and roles, e.g. $C(a)$ and $R(a, b)$, meaning, resp., that $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$.

Definition 2.1. *Given two concept descriptions C and D, C* subsumes *D, denoted by $C \sqsupseteq D$, iff for every interpretation $\mathcal{I}$ it holds that $C^{\mathcal{I}} \supseteq D^{\mathcal{I}}$.*

Example 2.1. An instance of concept definition in the proposed language is: $\mathsf{Father} \equiv \mathsf{Male} \sqcap \exists \mathsf{hasChild}.\mathsf{Person}$ which corresponds to the sentence: *"a father is a male (person) that has some persons as his children"*. The following are instances of simple assertions: $\mathsf{Male}(\mathsf{Leonardo})$, $\mathsf{Male}(\mathsf{Vito})$, $\mathsf{hasChild}(\mathsf{Leonardo}, \mathsf{Vito})$.

Supposing that $\mathsf{Male} \sqsubseteq \mathsf{Person}$ is known (in the T-Box), one can deduce that: $\mathsf{Person}(\mathsf{Leonardo})$, $\mathsf{Person}(\mathsf{Vito})$ and then $\mathsf{Father}(\mathsf{Leonardo})$.

Given these primitive concepts and roles, it is possible to define many other related concepts: $\mathsf{Parent} \equiv \mathsf{Person} \sqcap \exists \mathsf{hasChild}.\mathsf{Person}$ and $\mathsf{FatherWithoutSons} \equiv$

Table 1. $\mathcal{ALC}$ constructors and their meaning.

Name	Syntax	Semantics
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
concept	C	$C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
concept negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
concept conjunction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$
concept disjunction	$C_1 \sqcup C_2$	$C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$
existential restriction	$\exists R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}((x,y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}})\}$
universal restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}}((x,y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}})\}$

$\mathsf{Male} \sqcap \exists \mathsf{hasChild.Person} \sqcap \forall \mathsf{hasChild.}(\neg \mathsf{Male})$. It is easy to see that the following relationships hold: $\mathsf{Parent} \sqsupseteq \mathsf{Father}$ and $\mathsf{Father} \sqsupseteq \mathsf{FatherWithoutSons}$. □

A related inference used in the following is *instance checking*, that is deciding whether an individual is an instance of a concept [9]. Conversely, it may be necessary to solve the *realization problem* that requires finding the concepts which an individual belongs to, especially the most specific one, if any:

Definition 2.2. *Given an A-Box $\mathcal{A}$ and an individual a, the* most specific concept *of a w.r.t. $\mathcal{A}$ is the concept C, denoted $\mathsf{MSC}_{\mathcal{A}}(a)$, such that $\mathcal{A} \models C(a)$ and $C \sqsubseteq D$, $\forall D$ such that $\mathcal{A} \models D(a)$.*

In the general case of a cyclic A-Box expressed in a DL endowed with existential or numeric restriction the MSC cannot be expressed as a finite description [9], thus it can only be approximated. Generally an approximation of the MSC is considered up to a certain depth k. The maximum depth k has been shown to correspond to the depth of the considered A-Box [11].

Especially for rich DL languages such as $\mathcal{ALC}$, many semantically equivalent (yet syntactically different) descriptions can be given for the same concept. Nevertheless, equivalent concepts can be reduced to a normal form by means of rewriting rules that preserve their equivalence [9]:

Definition 2.3. *A concept description D is in* $\mathcal{ALC}$ normal form *iff $D \equiv \bot$ or $D \equiv \top$ or if $D = D_1 \sqcup \cdots \sqcup D_n$ ($\forall i = 1, \ldots, n$, $D_i \not\equiv \bot$) with*

$$D_i = \bigsqcap_{A \in \mathsf{prim}(D_i)} A \sqcap \bigsqcap_{R \in N_R} \left[\forall R.\mathsf{val}_R(D_i) \sqcap \bigsqcap_{E \in \mathsf{ex}_R(D_i)} \exists R.E \right]$$

where: $\mathsf{prim}(C)$ is the set of all (negated) primitives occurring at the top level of C; $\mathsf{val}_R(C)$ is the conjunction $C_1 \sqcap \cdots \sqcap C_n$ in the value restriction of role R, if any (otherwise $\mathsf{val}_R(C) = \top$); $\mathsf{ex}_R(C)$ is the set of concepts in the value restriction of the role R.
For any R, every sub-description in $\mathsf{ex}_R(D_i)$ and $\mathsf{val}_R(D_i)$ is in normal form.

3 A Dissimilarity Measure for $\mathcal{ALC}$

As a first step we need to define a dissimilarity measure for $\mathcal{ALC}$ descriptions. In order to achieve this goal, we introduce a function which is necessary for the correct definition of a dissimilarity measure. This should be a definite positive function on the set of $\mathcal{ALC}$ normal form concept description, defined making use of the syntax and semantics of the concepts (and roles) involved in the descriptions. The function is formally defined as follows:

Definition 3.1. *Let $\mathcal{L} = \mathcal{ALC}/_{\equiv}$ be the set of all concepts in $\mathcal{ALC}$ normal form and let $\mathcal{A}$ be an A-Box with canonical interpretation $\mathcal{I}$. f is a function $f : \mathcal{L} \times \mathcal{L} \mapsto R^+$ defined as follows:*
for all $C, D \in \mathcal{L}$, with $C = \bigsqcup_{i=1}^{n} C_i$ and $D = \bigsqcup_{j=1}^{m} D_j$

$$f(C,D) := f_{\sqcup}(C,D) = \begin{cases} 1 & \text{if } C \equiv D \\ 0 & \text{if } C \sqcap D \equiv \bot \\ \max\limits_{\substack{i=1,\ldots,n \\ j=1,\ldots,m}} f_{\sqcap}(C_i, D_j) & \text{otherwise} \end{cases}$$

$$f_{\sqcap}(C_i, D_j) := f_P(\mathsf{prim}(C_i), \mathsf{prim}(D_j)) + f_{\forall}(C_i, D_j) + f_{\exists}(C_i, D_j)$$

$$f_P(\mathsf{prim}(C_i), \mathsf{prim}(D_j)) := \frac{|(\mathsf{prim}(C_i))^{\mathcal{I}} \cup (\mathsf{prim}(D_j))^{\mathcal{I}}|}{|(\mathsf{prim}(C_i))^{\mathcal{I}} \cap (\mathsf{prim}(D_j))^{\mathcal{I}}|}$$

yet, $f_P(\mathsf{prim}(C_i), \mathsf{prim}(D_j)) = 0$ when $(\mathsf{prim}(C_i))^{\mathcal{I}} \cap (\mathsf{prim}(D_j))^{\mathcal{I}} = \emptyset$

$$f_{\forall}(C_i, D_j) := \sum_{R \in N_R} f_{\sqcup}(\mathsf{val}_R(C_i), \mathsf{val}_R(D_j))$$

$$f_{\exists}(C_i, D_j) := \sum_{R \in N_R} \sum_{k=1}^{N} \max_{p=1,\ldots,M} f_{\sqcup}(C_i^k, D_j^p)$$

where $C_i^k \in \mathsf{ex}_R(C_i)$ and $D_j^p \in \mathsf{ex}_R(D_j)$ and we suppose w.l.o.g. that $N = |\mathsf{ex}_R(C_i)| \geq |\mathsf{ex}_R(D_j)| = M$, otherwise the indices N and M are to be exchanged in the formula above.

The function f represents a measure of the overlap between two descriptions (namely C and D) expressed in $\mathcal{ALC}$ normal form. It is defined recursively beginning from the top level of the descriptions (a disjunctive level) up to the bottom level represented by (conjunctions of) primitive concepts.

In case of disjunction, the overlap between the two concepts is equal to the maximum of the overlaps calculated among all couples of disjuncts (C_i, D_j) that make up the top level of the considered concepts.

Then, since every disjunct is a conjunction of descriptions, it is necessary to calculate the overlap between conjunctive concepts. This is calculated as the

sum of the overlap among the parts that make up the conjunctive description. Specifically, a conjunctive form can have three different types of terms: primitive concepts, universal restrictions and existential restrictions. Since conjunction ($\sqcap$) is a symmetric operator, it is possible to put together every type of restriction (occurring at the same level) so it is possible to consider the conjunctions of primitive concepts, the conjunctions of existential restrictions and the conjunction of universal restrictions as specified in the definition of $\mathcal{ALC}$ normal form.

Next, the amount of the overlap for the three different type of conjunction is defined. Particularly, the amount of overlap between two conjunctions of (negated) primitive concepts is null if the they do not share any individual in their extension. Conversely, if the two concepts share some individual the overlap between them is computed as the ratio between the union and the intersection of their extensions which expresses how far the partial overlap is from the total overlap of the two concepts.

The computation of the overlap between, resp., descriptions expressed by universal and existential restrictions is a bit more complex. Considering the conjunction of universal restrictions, it is worthwhile to recall that every such restriction is a single conjunction linked by respect to a different role (since $\forall R.C \sqcap \forall R.D \equiv \forall R.(C \sqcap D)$). Moreover, the scope of each restriction is expressed in normal form. Thus, the amount of the overlap between two subconcepts (within C_i and D_j, resp.) that are scope of a universal restriction on a certain role R is given by the overlap between two concepts in normal form (computed by $f_{\sqcup}$); of course, if no disjunction occurs at the top level, it is possible to regard the concept description as a disjunction of a single term to which $f_{\sqcup}$ applies in a simple way. Since one may have a conjunction of concepts with universal restrictions, one per different role, the overlap of this conjunction is given by the sum of the overlap yielded by each restriction, rather than every restriction scope. Note that, when a universal restriction on a role occurs only in one of the descriptions, then the computation assumes $\top$ as the corresponding concept in the other description.

Now we turn to analyze the computation of the amount of the overlap between two descriptions made up of conjunctions of existential restrictions. For the dissimilarity between existential restrictions, we may recur to *existential mappings*. Supposing that $N = |\mathsf{ex}_R(C_i)| \geq M = |\mathsf{ex}_R(D_j)|$, such a mapping can be defined as a function $\alpha : \{1, \ldots, N\} \mapsto \{1, \ldots, M\}$. If each element of $\mathsf{ex}_R(C_i)$ and $\mathsf{ex}_R(D_j)$ is indexed with an integer in the ranges $[1, N]$ and $[1, M]$, resp., then any function α maps each concept description $C_i^k \in \mathsf{ex}_R(C_i)$ to $D_j^p \in \mathsf{ex}_R(D_j)$. Since each C_i^k (resp. D_j^p) is in normal form, it is possible to calculate the amount of their overlap using $f_{\sqcup}$. Fixed a role R and considered a certain C_i^k (with $k \in [1, N]$), the amount of the overlap between C_i^k and D_j^p (with $p \in [1, M]$) is computed. We are supposing that $N \geq M$, thus each existential restriction on role R is coupled with the one on the same role in other description scoring the maximum amount of overlap. These maxima are summed up per single role. In case of absence of role restrictions on a certain role from either description then it is considered as the concept $\top$.

Summing up, we have defined a measure whose baseline (counts on the extensions of primitive concepts) depends on the semantics of the knowledge base, as conveyed by the ABox assertions. This is in line with to the ideas in [10, 8], where semantics is elicited as a probability distribution over the domain of the interpretation Δ.

Now, it is possible to derive a dissimilarity measure based on f as follows

Definition 3.2. *Let $\mathcal{L}$ be the set of descriptions in $\mathcal{ALC}$ normal form and let f be an overlap function defined as above. The* dissimilarity measure *d is a function $d : \mathcal{L} \times \mathcal{L} \mapsto [0, 1]$ such that, for all $C = \bigsqcup_{i=1}^{n} C_i$ and $D = \bigsqcup_{j=1}^{m} D_j$ concept descriptions in $\mathcal{ALC}$ normal form:*

$$d(C, D) := \begin{cases} 1 & \text{if } f(C, D) = 0 \\ 0 & \text{if } f(C, D) = 1 \\ \frac{1}{f(C,D)} & \text{otherwise} \end{cases}$$

The function d measures the level of dissimilarity between two concepts, say C and D, in $\mathcal{ALC}$ normal form using the function f that expresses the amount of overlap between the two concepts. Particularly, if $f(C, D) = 0$ then this means that there is no overlap between the considered concepts, therefore d must indicate that the two concepts are totally different, indeed $d(C, D) = 1$ i.e. it amounts to the maximum value of its range. If $f(C, D) = 1$ this means that the two concepts are totally overlapped and consequently $d(C, D) = 0$ that means that the two concept are undistinguishable, indeed d assumes the minimum value of its range. If the considered concepts have a partial overlap then their dissimilarity is lower as much as the two concept are more overlapped, since in this case $f(C, D) > 0$ and consequently $0 < d(C, D) < 1$.

Let us recall that, for every individual in the A-Box, it is possible to calculate the most specific concept of an individual a w.r.t. an A-box, $\mathsf{MSC}(a)$ (see Def. 2.2) or at least its approximation $\mathsf{MSC}^k(a)$ up to a certain description depth k. In some cases these are equivalent concepts but in general $\mathsf{MSC}^k(a) \sqsupseteq \mathsf{MSC}(a)$. This notion is exploited to lift the individuals to the concept level.

Let a and b two individuals in a given A-Box. We can consider $A^* = \mathsf{MSC}^k(a)$ and $B^* = \mathsf{MSC}^k(b)$ (we also suppose that they are in $\mathcal{ALC}$ normal form). Now, in order to assess the dissimilarity between the considered individuals, the dissimilarity measure d can be applied to these descriptions, as follows:

$$d(a, b) := d(A^*, B^*) = d(\mathsf{MSC}^k(a), \mathsf{MSC}^k(b))$$

Analogously, the dissimilarity value between a concept description C and an individual a can be computed by determining the MSC approximation of the individual and then applying the dissimilarity measure:

$$\forall a : \ d(a, C) := d(\mathsf{MSC}^k(a), C)$$

This case may turn out to be particularly handy both in inductive reasoning (construction, repairing of knowledge bases) and in information retrieval.

We prove that d function actually is a dissimilarity measure (or *dissimilarity function* [12]), according to the following formal definition:

Definition 3.3. *Let* S *be a non empty set of elements. A* dissimilarity measure *for S is a real-valued function r defined on the set $S \times S$ that fulfills the properties:*
1. $r(a,b) \geq 0,\ \forall a,b \in S$ *(positive definiteness);*
2. $r(a,b) = r(b,a),\ \forall a,b \in S$ *(symmetry);*
3. $\forall a,b \in S\colon\ r(a,b) \geq r(a,a)$

Proposition 3.1. *The function d is a dissimilarity measure on $\mathcal{L} = \mathcal{ALC}/_{\equiv}$.*

Proof.
1. *trivial: by construction d computes dissimilarity by using sums of positive quantities and maxima computed on sets of such values.*
2. *by the commutativity of the sum and maximum operators.*
3. *by the definition of d, it holds that $d(C,C) = 0$ and $d(C,C') = 0$ if C is semantically equivalent to C'. In all other different cases, $\forall D \in \mathcal{L}$ and D not semantically equivalent to D ($C \not\equiv D$), we have: $d(C,D) > 0$* □

The computational complexity of our dissimilarity measure d is strictly related to that of f. The measure also relies on some reasoning services, namely subsumption and instance-checking, therefore its complexity depends on the complexity of these inferences too. In order to assess the complexity of d, we distinguish three different cases descending from being d based on $f_{\sqcup}$.
Let $C = \bigsqcup_{i=1}^{n} C_i$ and $D = \bigsqcup_{j=1}^{m} D_j$ be two descriptions in normal form:
– *Case 1: C and D are semantically equivalent.* Only subsumption is involved in order to verify the semantic equivalence of the concepts. Thus $\mathit{Compl}(d) = 2 \cdot \mathit{Compl}(\sqsupseteq)$, where $\mathit{Compl}(\cdot)$ and $\sqsupseteq$ represent, resp., complexity and subsumption;
– *Case 2: C and D are disjoint yet not semantically equivalent.* Subsumption and conjunction are involved. Being the time complexity of conjunction a constant, we have the same complexity of the previous case
– *Case 3: C and D are not semantically equivalent nor disjoint.* The complexity depends on the structure of the concepts. It is necessary to compute $f_{\sqcap}$ for $n \cdot m$ times; so the complexity is: $\mathit{Cmpl}(d) = nm \cdot \mathit{Cmpl}(f_{\sqcap}) = nm \cdot [\mathit{Cmpl}(f_P) + \mathit{Cmpl}(f_{\forall}) + \mathit{Cmpl}(f_{\exists})]$. Thus we analyze the complexity of f_P, $f_{\forall}$, $f_{\exists}$.
The dominant operation when computing f_P is instance checking (IC) used for determining the concept extensions. So we conclude that $C(f_P) = 2 \cdot C(IC)$. The computation of $f_{\forall}$ and $f_{\exists}$ apply recursively the definition of $f_{\sqcup}$ on less complex descriptions. A maximum of $|N_R|$ calls of $f_{\sqcup}$ are needed for computing $f_{\forall}$, while the calls of $f_{\sqcup}$ needed for $f_{\exists}$ are $|N_R| \cdot N \cdot M$, where $N = |\mathsf{ex}_R(C_i)|$ and $M = |\mathsf{ex}_R(D_j)|$ as in Def. 3.1. Summing up as in the previous equation:
$\mathit{Cmpl}(d) = nm \cdot [(2 \cdot \mathit{Cmpl}(IC)) + (|N_R| \cdot \mathit{Cmpl}(f_{\sqcup})) + (|N_R| \cdot M \cdot N \cdot \mathit{Cmpl}(f_{\sqcup}))]$

We conclude that the complexity of the computation of d depends on the complexity of the instance-checking for $\mathcal{ALC}$ which is P-space [9]. Nevertheless, in practical applications, these computations may be efficiently carried out exploiting the statistics that are maintained by the DBMSs query optimizers. Besides, the counts that are necessary for computing the concept extensions could be estimated by means of the probability distribution over the domain.

4 Conclusions and Further Developments

Similarity measures turn out to be useful in several tasks such as, classification, case-based reasoning, clustering, etc. A novel dissimilarity measure d has been introduced, derived from the measure f of the overlap between $\mathcal{ALC}$ descriptions, and based on the underlying semantics based on ABox interpretation.

We have also shown how to apply this function to measuring the dissimilarity between individuals and also a individual-concept dissimilarity, which may be more useful in knowledge discovery tasks.

In particular, defining a measure that is applicable for both the concepts to individual similarity and between individuals one, it is suitable for agglomerative clustering and for divisional clustering too. A further investigation will concern the derivation of a distance measure, which amounts to finding a measure that fulfils the triangular property.

These ideas are being exploited also for defining kernels on rich representations like DLs, thus allowing the exploitation of the efficiency of SVMs and the other related methods.

References

[1] Jang, J., Conrath, D.: Semantic symilarity based on corpus statistic and lexical taxonomy. In: Proceedings of the International Conference on Computational Linguistics. (1997)

[2] Guarino, N., Masolo, C., Verete, G.: Ontoseek: Content-based access to the web. IEEE Intelligent Systems **3** (1999) 70–80

[3] Bright, M.W., Hurson, A.R., Pakzad, S.H.: Automated resolution of semantic heterogeneity in multidatabases. ACM Transaction on Database Systems **19** (1994) 212–253

[4] Tversky, A.: Features od similarity. Psycological Review **84** (1997) 327–352

[5] Resnik, P.: Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. Journal of Artificial Intelligence Research **11** (1999) 95–130

[6] Weinstein, P., Bimingham, P.: Comparing concepts in differentiated ontologies. In: Proceedings of 12th Workshop on Knowledge Acquisition, Modelling, and Management. (1999)

[7] Rodríguez, M., Egenhofer, M.: Determining semantic similarity among entity classes from different ontologies. IEEE Transaction on Knowledge and Data Engineering **15** (2003) 442–456

[8] Borgida, A., Walsh, T., Hirsh, H.: Towards measuring similarity in description logics. In: Working Notes of the International Description Logics Workshop. CEUR Workshop Proceedings, Edinburgh, UK (2005)

[9] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: The Description Logic Handbook. Cambridge University Press (2003)

[10] Bacchus, F.: Lp, a logic for representing and reasoning with statistical knowledge. Computational Intelligence **6** (1990) 209–231

[11] Mantay, T.: Commonality-based ABox retrieval. Technical Report FBI-HH-M-291/2000, Dept. of Computer Science, University of Hamburg, Germany (2000)

[12] Bock., H.: Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data. Springer-Verlag (1999)

Deducing a Term Taxonomy from Term Similarities

Georges Dupret and Benjamin Piwowarski

Computer Science Department, University of Chile.
gdupret@dcc.uchile.cl
bpiwowar@dcc.uchile.cl

Abstract. We show that the singular value decomposition of a term similarity matrix induces a term taxonomy. This decomposition, used in Latent Semantic Analysis and Principal Component Analysis for text, aims at identifying "concepts" that can be used in place of the terms appearing in the documents. Unlike terms, concepts are by construction uncorrelated and hence are less sensitive to the particular vocabulary used in documents. In this work, we explore the relation between terms and concepts and show that for each term there exists a latent subspace dimension for which the term coincides with a concept. By varying the number of dimensions, terms similar but more specific than the concept can be identified, leading to a term taxonomy.

Keywords: Taxonomy, principal component analysis, latent semantic analysis, information retrieval, eigenvalue decomposition.

1 Introduction

Automated management of digitalized text requires a computer representation of the information. A common method consists in representing documents by a bag-of-words or set of features, generally a subset of the terms present in the documents. This gives rise to the vector space model where documents are points in an hyperspace with features as dimensions: The more important a feature in a document, the larger the coordinate value in the corresponding dimension [15].

Clearly, much information is lost when discarding the term order but the more significant limitation is that only the presence and the co-occurrence of terms are taken into account, not their meaning. Consequently, synonyms appear erroneously as distinct features and polysemic terms as unique features. This serious limitation is an avatar of the feature independence assumption implicit in the vector representation.

In the more general *statistical models* [27] (OKAPI) representations of queries and documents are clearly separated. Relevance of a document to a query is estimated as the product of individual term contributions [4]. The corresponding assumption is not much weaker than strict independence.

Term dependence is taken into account in Language Models like *n-grams* and their applications to Information Retrieval [24], but generally within windows of

two or three terms. The Bayesian network formalism [6, 26] also allows for term dependence, but their application to a large number of features is unpractical.

Principal Component Analysis (PCA) [7] for text (and the related Latent Semantic Analysis method) offers a different approach: Uncorrelated linear combinations of features –the latent "concepts"– are identified. The lack of correlation is taken to be equivalent to independence as a first approximation, and the latent "concepts" are used to describe the documents. In this paper, we show that more than a list of uncorrelated latent concepts, Principal Component Analysis uncovers a hierarchy of terms that share a "*related and more specific than*" relation. Taking advantage of this structure to estimate a similarity measure between query and documents is a future topic of work, while applications of taxonomy to Information Retrieval are presented in Sect. 6.

This paper extends the results of [8] beyond the context of Latent Semantic Analysis and PCA to all type of symmetric similarity measures between terms and hence documents. It proposes a theoretical justification of the results presented there, as well as a geometric interpretation in Sect. 4. The main contribution, a method to derive a taxonomy, is presented in Sect. 3. Numerical experiments in Sect. 5 validate the method while a review of automatic taxonomy generation methods is proposed in Sect. 6.

2 Term Similarity Measure

The estimation of the similarity between terms in Information Retrieval is generally based on term co-occurrences. Essentially, if we make the assumption that each document of a collection covers a single topic, two terms that co-occur frequently in the documents necessarily refer to a common topic and are therefore somehow similar. If the documents are not believed to refer to a single topic, it is always possible to divide them into shorter units so that the hypothesis is reasonably verified.

2.1 Measures of Similarity

The Pearson correlation matrix $\mathbf{S}$ associated to the term by document matrix $\mathbf{A}$ is a common measure of term similarity. It reflects how systematically terms co-occur: if they always occur together, their correlation will be one, if they occur only by chance, their correlation will be close to zero, and if they never co-occur, the presence of one of the term will be a sure sign that the other is absent, and their correlation will be minus one. To estimate it, we first compute the "mean document" $\overline{\mathbf{d}}$ of the mean term weights, the standard deviation $\mathrm{sd}(t_j)$ of each term weight and the normalized version of documents:

$$a'_{i,j} = \frac{(a_{i,j} - \overline{\mathbf{d}}_j)}{\mathrm{sd}(t_j)}$$

where $a_{i,j}$ is the (i, j) element of $\mathbf{A}$. The $a'_{i,j}$ form a normalized document matrix $\mathbf{A}_\mathrm{N}$ from which we deduce the correlation matrix: $\mathbf{S} = 1/(D-1)\mathbf{A}_\mathrm{N}^T\mathbf{A}_\mathrm{N}$.

Other metrics are possible. Nanas et al. [21] count the number of term co-occurrence in sliding windows of fixed length, giving more weight to pairs of terms appearing close from each other. Park et al. [23] use a Bayesian network. Crestani in [5] proposes an *Expected Mutual Information Measure* inspired from Information Theory.

The method we present here does not rely on a particular measure of similarity or distance. The only requirement is an estimate of the similarity between any two index terms, represented by a symmetric matrix $\mathbf{S}$.

2.2 Similarity in Lower Dimensional Space

In the vector space representation of documents, index terms correspond to the base vectors of an hyperspace where documents are represented by points. If to each term j corresponds a base vector $\mathbf{e}_j$, an arbitrary document d is represented by $\mathbf{d} = \sum_{j=1}^{N} \omega_j \mathbf{e}_j$ where ω_j is the weight of term j in the document d. Weights are usually computed using the well known *tf.idf* [28] formula and lay between 0 and 1. The inconvenient of this representation stems from the implicit assumption of independence between terms: Consider two documents d_a and d_b each composed of a different single term. Independently of whether the single terms are synonyms, unrelated or antonyms, the document similarity in the hyperspace representation will be null because they coincide with two different base vectors. A more desirable result would be the measure of similarity between terms u and v like the element (u, v) of matrix $\mathbf{S}$. This can be achieved by defining a new similarity measure between documents: The dot product in base $\mathbf{S}$ between the normalized document vectors[1].

$$\frac{\mathbf{d}_a^T}{|\mathbf{d}_a|}\mathbf{S}\frac{\mathbf{d}_b}{|\mathbf{d}_b|} = S_{u,v}$$

Alternatively, we can define an *extended representation* of a document d as $(1/|\mathbf{d}|)\mathbf{d}^T\sqrt{\mathbf{S}}$ and use the traditional cosine similarity measure[2].

The idea of introducing the similarity between terms to compute document similarity is closely related to Latent Semantic Analysis and Principal Component Analysis for text [8]. In the last method, the similarity between documents and a query is computed as $\mathbf{r}(k) = \mathbf{A}^T\mathbf{S}(k)\mathbf{q}$ where $\mathbf{A}$ is the matrix formed by the space vector representation of documents presented in Sect. 2.1, q is a query and $r_i(k)$, the i^{th} component of $\mathbf{r}(k)$, is the similarity of document i with the query. The analogy with the *extended document representation* is clear, but instead of using the original similarity matrix $\mathbf{S}$, we use the rank k approximation of its eigenvalue decomposition. The matrix $\mathbf{S}$ can be decomposed into a product including the orthonormal matrix $\mathbf{V}$ and the diagonal matrix $\mathbf{\Sigma}$ of its eigenvalues σ_ℓ in decreasing value order: $\mathbf{S} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$. The best approximation following the Frobenius norm of the matrix $\mathbf{S}$ in a subspace of dimensionality

[1] $\mathbf{S}$ being symmetric, but not necessarily full rank, this dot product introduces a quasi-norm [12]

[2] $\sqrt{\mathbf{S}}$ always exists because the singular values of $\mathbf{S}$ are all positive or null.

$k < N$ is obtained by setting to zero the eigenvalues σ_ℓ for $\ell > k$. Noting $\mathbf{V}(k)$ the matrix composed of the k first columns of $\mathbf{V}$ and $\mathbf{\Sigma}(k)$ the diagonal matrix of the first k eigenvalues, we have $\mathbf{S}(k) = \mathbf{V}(k)\mathbf{\Sigma}(k)\mathbf{V}(k)^T$.

We can now represent in extended form a document $\mathbf{t}_u$ formed of a unique index term u in the rank k approximation of the similarity matrix:

$$\begin{aligned} \mathbf{t}_u^T &= \mathbf{e}_u^T \sqrt{\mathbf{S}} \\ &= \mathbf{e}_u^T \mathbf{V}(k)\mathbf{\Sigma}(k)^{1/2} \\ &= \mathbf{V}_{u,\cdot}(k)\mathbf{\Sigma}(k)^{1/2} \end{aligned} \tag{1}$$

where $\mathbf{\Sigma}(k)^{1/2}$ is the diagonal matrix of the square root of the eigenvalues in decreasing order and $\mathbf{V}_{u,\cdot}(k)$ is the u^{th} row of $\mathbf{V}(k)$. Following the terminology introduced by Latent Semantic Analysis, the columns of $\mathbf{V}(k)$ represent latent concepts. The documents in general as well as the single term documents are represented with minimal distortion[3] as points in the k dimensional space defined by the k first columns – i.e. the eigenvectors – of $\mathbf{V}$ instead of the N dimensional space of index terms. This is possible only if the selected eigenvectors summarize the important features of the term space, hence the idea that they represent latent concepts.

In the next sections, we analyze the properties of the rank k approximation of the similarity matrix for different ranks and show how a taxonomy can be deduced.

3 Identification of the Concepts of a Term

We explore in this section the relation between terms and concepts. Sending a similarity matrix onto a subspace of fewer dimensions implies a loss of information. We will see that it can be interpreted as the merging of terms meanings into a more general concept that encompasses them. As an example, "cat" and "mouse" are specializations of the concept of "mammal," which itself is subsumed by "animal." The results we obtain are not as neat, but this example illustrates our objective.

We first examine the conditions under which a term coincides with a concept. Then we use the results to deduce a taxonomy.

3.1 Term Validity Rank

A similarity matrix row $\mathbf{S}_{j,\cdot}$ and its successive approximations $\mathbf{S}(k)_{j,\cdot}$ represent a single term document $\mathbf{t}_j$ in terms of its similarity with all index terms. We seek a representation that is sufficiently detailed or encompass enough information for the term to be correctly represented. A possible way is to require that a single term document is more similar to itself than to any other term document:

[3] according to the Frobenius norm

Definition 1 (Validity). *A term is correctly represented in the k-order approximation of the similarity matrix only if it is more similar to itself than to any other term. The term is then said to be valid at rank k.*

If we remember that the normalized single term documents correspond to the base vectors, $\mathbf{e}_u$, the definition of validity requires: $\mathbf{e}_u^T\mathbf{S}(k)\mathbf{e}_u > \mathbf{e}_u^T\mathbf{S}(k)\mathbf{e}_v \; \forall u \neq v$, or $\mathbf{t}_u^T\mathbf{t}_u > \mathbf{t}_u^T\mathbf{t}_v \; \forall u \neq v$. This is verified if the diagonal term u of $\mathbf{S}$ is larger than any other element of the same column, i.e. if $\mathbf{S}(k)_{u,u} > \mathbf{S}(k)_{u,v} \quad \forall v \neq u$. In other words, even though the diagonal element corresponding to term i is not equal to unity –which denotes conventionally perfect similarity, it should be greater than the non-diagonal elements of the same row[4] to be correctly represented.

It is useful to define the rank below which a term ceases to be valid:

Definition 2 (Validity Rank). *A term `t` is optimally represented in the k-order approximation of the similarity matrix if it is valid at rank k and if $k-1$ is the largest value for which it is not valid. Rank k is the validity rank of term `t` and is denoted* rank(`t`).

In practice it might happen for some terms that validity is achieved and lost successively for a short range of ranks. It is not clear whether this is due to a lack of precision in the numerically sensitive eigenvalue decomposition process or to theoretical reasons.

The definition of validity was experimentally illustrated in [8] where all the documents containing a specific term `a` were replicated in the database with `a` replaced by some new term `a'`. The query composed of the term `a` was shown to return in alternation `a` and `a'` versions of the documents as long as the rank k of the approximation was below the validity rank of `a`. Beyond the validity rank, version of the documents containing the term `a` were returned first, suggesting that the representation of that term was ambiguous below rank(`a`), and unambiguous beyond it. This shows that if the rank of the similarity approximation matrix and the validity rank of a term used as a single word query coincide, then retrieval precision[5] is optimal. This justifies Definition 1 a posteriori. An extension to more than one term queries showed mixed results in [9]. A theoretical justification of the validity rank was presented recently in [2].

3.2 Term Taxonomy

In the experiment described above, we observed that terms `a` and `a'` were not correctly distinguished in the k-dimensional latent concept space if k is inferior to the validity rank of `a`[6]. This shows that 1) the two terms bear a common meaning to a certain extent, 2) the common meaning is more general than the

[4] $\mathbf{S}(k)$ is symmetric and the condition can be applied indifferently on rows or columns.
[5] We refer to the traditional definition of precision and recall.
[6] Terms `a` and `a'` being perfect synonyms, they have the same validity rank.

meaning of any of the two terms. For these two reasons, we call the common meaning the *concept*[7] shared by the two terms.

Moreover, we know by Definition 2 that below their validity rank, `a` and `a'` are more similar to some other terms than to themselves. If they are both more similar to a common term `c` valid at rank k, the representation of this term better covers the concept common to `a` and `a'`: We say that `a` and `a'` share the common concept $\mathtt{c}^*$ where the notation $\mathtt{c}^*$ is used to recall the difference between the representation of the single term document at full rank and at its validity rank.

Definition 3 (Concept of a Term). *A concept* $\mathtt{c}^*$ *associated to term* `c` *is a concept of term* $\mathtt{a}$ *if* $\text{rank}(\mathtt{c}) < \text{rank}(\mathtt{a})$ *and if for some rank* k *such that* $\text{rank}(\mathtt{c}) \leq k < \text{rank}(\mathtt{a})$, $\mathtt{a}^*$ *is more similar to* $\mathtt{c}^*$ *than to itself.*

The requirement that $\text{rank}(\mathtt{c}) < \text{rank}(\mathtt{a})$ ensures that $\mathtt{a}^*$ is never a concept of $\mathtt{c}^*$ if $\mathtt{c}^*$ is a concept of $\mathtt{a}^*$. If we associate terms to nodes and add directed links from the terms to their concepts, we obtain a directed acyclic graph (DAG). In practice, there is a whole range of ranks between $\text{rank}(\mathtt{c})$ and $\text{rank}(\mathtt{a})$ where concept $\mathtt{a}^*$ points to its concept $\mathtt{c}^*$, and we keep only the largest one to construct the graph. By identifying the concepts associated to all the terms, we can construct a taxonomy. This is illustrated in Section 5.

4 Geometrical Interpretation

We have seen that single term documents $\mathbf{e}_u$ are sent to $\mathbf{t}_u$ in the subspace of latent concepts through the operation described in Eq. 1: $\mathbf{t}_u^T = \mathbf{V}_{u,\cdot}(k)\mathbf{\Sigma}(k)^{1/2}$. Example of such single term documents are represented in Fig. 4 on the plane defined by the first two eigenvectors. Documents containing several terms can be understood as linear combinations of the single term documents with coefficients $\omega_j \geq 0$ and $\sum_j \omega_j = 1$: $\mathbf{d} = \sum_j \omega_j \mathbf{e}_j$ in the original term space leads to a representation $\sum_j \omega_j \mathbf{t}_j$ in the latent concept space with the same ω_j. For example, a document composed of terms $\mathbf{t}_1$ and $\mathbf{t}_3$ in Fig. 4 (left) lies on the segment between these two points. Similarly, a document containing more terms will lie inside the polygon defined by the terms of the document. In conclusion, the envelope of the points cover all the accessible points, and hence represents the set of all the possible documents that can be expressed with the vocabulary of indexed terms. This can be interpreted as the universe of discourse.

A term is valid at rank k if, following Def. 1, we have for all terms $v \neq u$ that $\mathbf{t}_u^T \cdot \mathbf{t}_u > \mathbf{t}_u^T \cdot \mathbf{t}_v$. This is verified if

$$\mathbf{t}_u^T \cdot \mathbf{t}_u > \mathbf{t}_u^T \cdot \mathbf{x}$$

for all vector $\mathbf{x}$ of dimension k. This inequation defines a boundary hyper-plane passing through point $\mathbf{t}_u$ and perpendicular to the vector from the origin to this

[7] This concept differs from the notion of *latent concept* popularized by Latent Semantic Analysis.

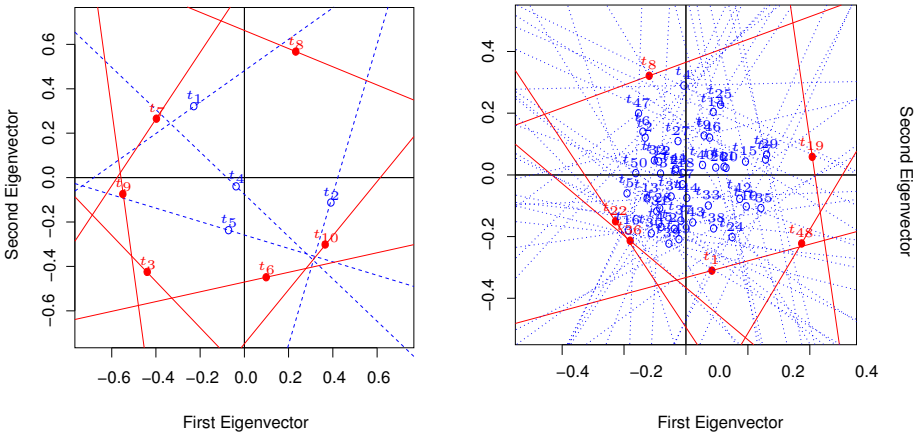


Fig. 1. Solid (red) circles represent valid terms, while hollow (blue) circles represent dominated terms. Continuous lines are the boundary hyper-planes of valid terms, while dotted lines are associated to dominated terms.

point. All points beyond this hyper-plane *dominate* the term $\mathbf{t}_u$ in the sense that $\mathbf{t}_u$ is more similar to them than to itself. In Fig. 4 (left), term $\mathbf{t}_1$ is dominated by $\mathbf{t}_7$ and term $\mathbf{t}_5$ is dominated by $\mathbf{t}_3$ and $\mathbf{t}_6$.

None of the invalid terms $\mathbf{t}_1$, $\mathbf{t}_4$ or $\mathbf{t}_5$ contribute to extend the universe of discourse; their meaning can be represented by a linear combination of the valid terms. In this respect, $\mathbf{t}_2$ is an exception because it lies outside the envelope of valid points. On the other hand, the frequency of such exceptions diminishes when the universe of discourse contain more terms because the polygon defined by the boundary hyper-planes tends to the envelope as the number of terms increases. Fig. 4 (right) illustrates this fact with as few as 50 terms.

In conclusion, the valid terms approximate the envelope of the universe of discourse at a given rank k, and all documents can be represented by a combination of the terms valid at that rank, but for some exceptions that become marginal as the number of index terms increases.

5 Numerical Experiments

Numerical experiments are based on the REUTERS collection [19]. This database gathers 21,578 articles extracted from the Reuters newswire in 1987. As a form of pre-processing, we extracted the stem of the words using the Porter algorithm [25][8], we removed stopwords and terms appearing in either more or less than two pre-specified thresholds. This resulted in a vocabulary of 2035 terms. We then mapped documents to vectors using with the traditional *tf.idf* weighting and computed the correlation matrix to estimate terms similarity.

[8] To facilitate reading, we refer to terms by one of their base forms even though only the stems are used in the experiment.

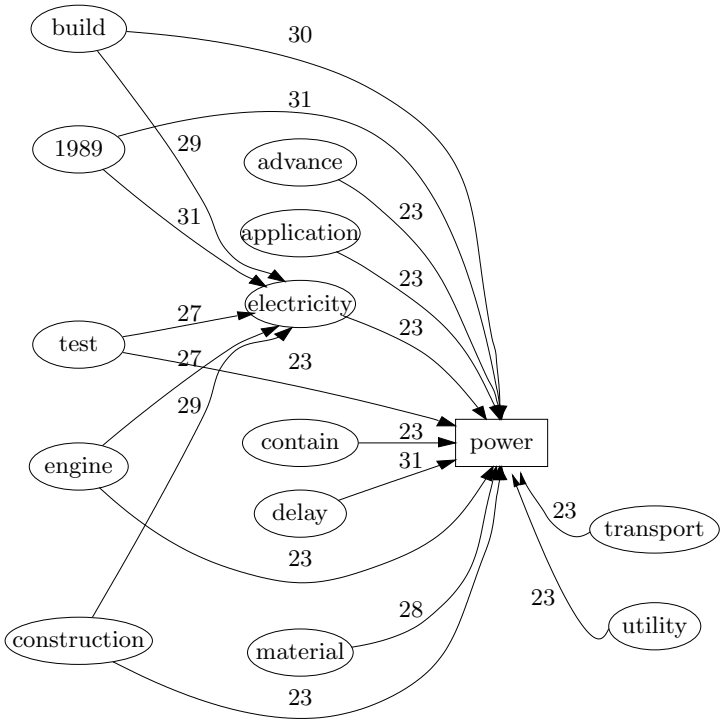


Fig. 2. Terms having `power` as a concept. Numbers represent the largest rank for which the link exists.

There are no standard procedures to evaluate hierarchies although some attempts have been proposed [17]. The comparison with a well established hierarchy like Wordnet is also problematic because they do not share the same objective. Wordnet is a lexical database for the English language while the taxonomy we present here necessarily reflects the document collection it is extracted from.

The visualization of the complete taxonomy is unpractical because of the large size of the vocabulary. We therefore discuss two representative subgraphs. Figure 2 shows all the terms pointing to a given context –`power`– while all the concepts of a given term –`gulf`– are shown in Fig. 3. Some associations are semantic: For example `utility`, `electricity` and `engine` clearly share a common meaning with `power`. Other associations are more contextual like `1989` that appears because most of the news about electric power appeared during 1989. Figure 3 also shows purely contextual associations. It represents all the concepts of the term `gulf` and their in-links. Terms having `gulf` as a concept have not been represented. This part of the graph alludes to a row in 1987 between the United States and Iran, the former accusing Tehran to prepare attacks against Kuwaiti petrol tankers (see News nbr. 8675 for example).

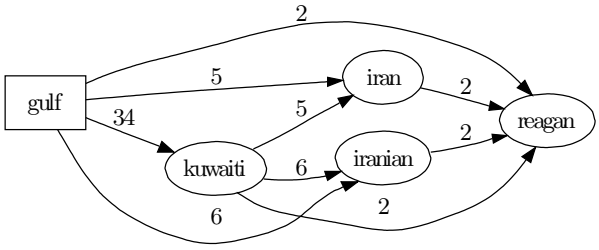


Fig. 3. Taxonomy of `gulf` towards its associated concepts. Concepts of concepts are not represented. Numbers represent the largest rank for which the link exists.

In general, terms representing concrete concepts, like `software` or `maize` lead to more directly interpretable graphs than terms like `allow` or `want`. Although the initial term selection from the corpus was particularly rough –we just kept terms appearing in a range of document frequencies– the taxonomy produced is surprisingly appealing. It is also expected that the use of a larger collection would further enhance the results: The larger the number of topics and the number of different documents, the lower the importance of contextual associations.

6 Related Work

Taxonomy find numerous applications in Information Retrieval: They are used to search information interactively [10], for query expansion [23], for interactive query expansion [14], etc. Most of them are constructed manually or semi-automatically.

Different fully automatically methods have been proposed. Sanderson and Croft [29] use the co-occurrence information to identify a term that subsumes other terms. More specifically, a term, u, is said to subsume another term, v, if the documents which v occurs in are a subset of the documents in which u occurs. Given that a more frequent term tends to be more general [30], Subsumption Hierarchies organize terms in a 'general to specific' manner. The method proposed by Nanas et al. [21] is similar, but a subsumption relation is accepted if the terms involved are also correlated. The correlation is measured for terms appearing in windows of fixed length, and depends on the distance between them.

Hyponymy relations are derived from lexico-syntactic rules rather than plain co-occurrence in [13]. Another approach is to rely on frequently occurring words within phrases or lexical compounds. The creation of such Lexical Hierarchies has been explored by [1, 22] and compared with Subsumption Hierarchies in [17]. In addition to the above two approaches, Lawrie et al. have investigated the generation of a concept hierarchy using a combination of a graph theoretic algorithm and a language model [18].

While Sanderson and Croft [29] consider only a subsuming/subsumed relation between terms, Glover et al. [11] base their hierarchy discovering algorithm

on three categories: If a term is very common in a cluster of documents, but relatively rare in the collection, then it may be a good "self" term. A feature that is common in the cluster, but also somewhat common in the entire collection, is a description of the cluster, but is more general and hence may be a good "parent" feature. Features that are common in the cluster, but very rare in the general collection, may be good "child" features because they only describe a subset of the positive documents.

Traditional data mining and machine learning methods have been attempted also. In [20], the learning mechanism is based on the Srikant and Agrawal [31] algorithm for discovering generalized association rules. A Bayesian network approach is proposed in [23]. Another graph-based representation of structural information combined with a substructure discovery technique is the base of the SUBDUE algorithm presented in [16]. Hierarchical clustering algorithm can be used to derive relations between terms, but, as observed in [3] cluster labelling is a challenging activity.

7 Conclusion

We used geometrical arguments to show that the projection of any document in the k dimensional Eigenspace of the similarity matrix can be represented by a subset of the valid concepts, independently of the terms present in the document. In this sense, the valid concepts are a sufficient representation all the semantic that can be expressed in the specified dimension.

We also showed that the term similarity matrix induces a hierarchical relation among the concepts associated to the terms and we illustrated the construction of such a taxonomy on the REUTERS newswire collection. This suggests that all the information obtained by the eigenvalue decomposition process is not exploited when sending documents to a lower dimensional space before estimating the cosine distances between the query and the documents. The derivation of a distance measure that takes the hierarchical structure into account is a topic for future work.

References

1. P. G. Anick and S. Tipirneni. The paraphrase search assistant: terminological feedback for iterative information seeking. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 153–159, New York, NY, USA, 1999. ACM Press.
2. H. Bast and D. Majumdar. Understanding spectral retrieval via the synonymy graph. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. Max-Planck-Institut fr Informatik, 2005.
3. S.-L. Chuang and L.-F. Chien. A practical web-based approach to generating topic hierarchy for text segments. In *CIKM '04: Proceedings of the Thirteenth ACM conference on Information and knowledge management*, pages 127–136, New York, NY, USA, 2004. ACM Press.

4. W. S. Cooper. Some inconsistencies and misidentified modelling assumptions in probabilistic information retrieval. In N. J. Belkin, P. Ingwersen, and A. M. Pej, editors, *Proceedings of the 14th ACM SIGIR*, Copenhagen, Danemark, 1992. ACM Press.
5. F. Crestani, I. Ruthven, M. Sanderson, and C. van Rijsbergen. The troubles with using a logical model of ir on a large collection of documents. experimenting retrieval by logical imaging on trec. in proceedings of the fourth text retrieval conference (trec-4), 1995. In *Proceedings of the Fourth Text Retrieval Conference (TREC-4)*, 1995.
6. L. M. de Campos, J. M. Fernndez-Luna, and J. F. Huete. The bnr model: foundations and performance of a bayesian network-based retrieval model. *International Journal of Approximate Reasoning*, 34:265–285, 2003.
7. S. Deerwester, S. Dumais, G. Furnas, and T. Landauer. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41:391–407, 1990.
8. G. Dupret. Latent concepts and the number orthogonal factors in latent semantic analysis. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 221–226. ACM Press, 2003.
9. G. Dupret. Latent semantic indexing with a variable number of orthogonal factors. In *Proceedings of the RIAO 2004, Coupling approaches, coupling media and coupling languages for information retrieval*, pages 673–685. Centre de Hautes Etudes Internationales d'informatique documentaire, C.I.D., April 26-28 2004.
10. R. Feldman, M. Fresko, Y. Kinar, Y. Lindell, O. Liphstat, M. Rajman, Y. Schler, and O. Zamir. Text mining at the term level. In *Principles of Data Mining and Knowledge Discovery*, pages 65–73, 1998.
11. E. Glover, D. M. Pennock, S. Lawrence, and R. Krovetz. Inferring hierarchical descriptions. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 507–514, New York, NY, USA, 2002. ACM Press.
12. D. Harville. *Matrix Algebra from a Statistician's Perspective*. Springer-Verlag, New York, 1997. 14.
13. M. A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics*, pages 539–545, Morristown, NJ, USA, 1992. Association for Computational Linguistics.
14. H. Joho, C. Coverson, M. Sanderson, and M. Beaulieu. Hierarchical presentation of expansion terms. In *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*, pages 645–649, New York, NY, USA, 2002. ACM Press.
15. W. P. Jones and G. W. Furnas. Pictures of relevance: a geometric analysis of similarity measures. volume 38, pages 420–442, New York, NY, USA, 1987. John Wiley & Sons, Inc.
16. I. Jonyer, D. J. Cook, and L. B. Holder. Graph-based hierarchical conceptual clustering. *J. Mach. Learn. Res.*, 2:19–43, 2002.
17. D. Lawrie and W. Croft. Discovering and comparing topic hierarchies. In *Proceedings of RIAO 2000*, 2000.
18. D. Lawrie, W. B. Croft, and A. Rosenberg. Finding topic words for hierarchical summarization. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 349–357, New York, NY, USA, 2001. ACM Press.
19. D. Lewis. Reuters-21578 text categorization test collection, distribution 1.0. AT&T Labs-Research, 1997.

20. A. Maedche and S. Staab. Discovering conceptual relations from text. pages 321–325, 2000.
21. N. Nanas, V. Uren, and A. D. Roeck. Building and applying a concept hierarchy representation of a user profile. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 198–204, New York, NY, USA, 2003. ACM Press.
22. C. G. Nevill-Manning, I. H. Witten, and G. W. Paynter. Lexically-generated subject hierarchies for browsing large collections. In *International Journal on Digital Libraries*, volume 2(2-3), pages 111–123, 1999.
23. Y. C. Park, Y. S. Han, and K.-S. Choi. Automatic thesaurus construction using bayesian networks. In *CIKM '95: Proceedings of the fourth international conference on Information and knowledge management*, pages 212–217, New York, NY, USA, 1995. ACM Press.
24. J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, New York, NY, USA, 1998. ACM Press.
25. M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
26. B. A. N. Ribeiro and R. Muntz. A belief network model for ir. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260, New York, NY, USA, 1996. ACM Press.
27. S. Robertson and K. S. Jones. Simple proven approaches to text retrieval. Technical report tr356, Cambridge University Computer Laboratory, 1997.
28. G. Salton and C. Buckley. Automatic structuring and retrieval of large text files. *Communications of the ACM*, 32(2):97–107, 1994.
29. M. Sanderson and B. Croft. Deriving concept hierarchies from text. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–213, New York, NY, USA, 1999. ACM Press.
30. K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972. (Reprinted in Griffith, B. C. (Ed.) Key Papers in Information Science, 1980, and in Willett, P. (Ed.) Document Retrieval Systems, 1988).
31. R. Srikant and R. Agrawal. Mining generalized association rules. *Future Generation Computer Systems*, 13(2–3):161–180, 1997.

Semi-automatic construction of topic ontology

Blaž Fortuna[1], Dunja Mladenič[1], and Marko Grobelnik[1]

Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia,
blaz.fortuna@ijs.si, dunja.mladenic@ijs.si, marko.grobelnik@ijs.si
WWW home page: http://kt.ijs.si/

Abstract. In this paper, we review two techniques for topic discovery in collections of text documents (Latent Semantic Indexing and K-Means clustering) and present how we integrated them into a system for semi-automatic topic ontology construction. The system offers supports to the user during the construction process by suggesting topics and analysing them in real time.

1 Introduction

When working with large corpora of documents it is hard to comprehend and process all the information contained in them. Standard text mining and information retrieval techniques usually rely on word matching and do not take into account the similarity of words and the structure of the documents within the corpus. We try to overcome that by automatically extracting the topics covered within the documents in the coprus and helping the user to organize them into a topic ontology.

Topic ontology is a set of topics connected with different types of relations. Each topic includes a set of related documents. Construction of such an ontology from a given corpus can be a very time consuming task for the user. In order to get a feeling on what the topics in the corpus are, what the relations between topics are and, at the end, to assign each document to some certain topics, the user has to go through all the documents. We tried to overcame this by building a special tool which helps the user by suggesting the possible new topics and visualizing the topic ontology created so far - all in real time. This tool in combination with the corpus visualization tools [8] aims at assisting the user in a fast semi-automatic construction of the topic ontology from a large document collection.

We chose two different approaches for discovering topics within the corpora. The first approach is a linear dimensionality reduction technique, known as Latent Semantic Indexing (LSI) [5]. This technique relies on the fact that words related to the same topic co-occur together more often than words related to the different topics. The result of LSI are fuzzy clusters of words each describing one topic. The second approach we used for extracting topics is a well known k-means clustering algorithm [12]. It partitions the corpus into k clusters so that two documents within the same cluster are more closely related than two documents from two different clusters. We used this two algorithms for automatic suggestion of topics during the construction of the topic ontology.

This paper is organized as follows. Section 2 gives a short overview of the related work on building ontologies. Section 3 gives an introduction to the text mining techniques we used. Details about our system are presented in Section 4, followed by the conclusions in Section 5.

2 Related work on building ontologies

Different approaches have been used for building ontologies, most of them using mainly manual methods. An approach to building ontologies was set up in the CYC project [6], where the main step involved manual extraction of common sense knowledge from different sources. There have been some definitions of methodology for building ontologies, again assuming manual approach. For instance, the methodology proposed in [19] involves the following stages: identifying the purpose of the ontology (why to build it, how will it be used, range of the users), building the ontology, evaluation and documentation. The building of the ontology is further divided in three steps. The first is ontology capture, where key concepts and relationships are identified, a precise textual definition of them is written, terms to be used to refer to the concepts and relations are identified, the involved actors agree on the definitions and terms. The second step involves coding of the ontology to represent the defined conceptualization in some formal languages (committing to some meta-ontology, choosing a representation language and coding). The third step involves possible integration with existing ontologies. An overview of the methodologies for building ontologies is provided in [7], where several methodologies, including the above described one, are presented and analyzed against the IEEE Standard for Developing Software Life Cycle Processes viewing ontologies as parts of some software product.

Recently, a number of workshops at Artificial Intelligence and Machine Learning conferences (ECAI, IJCAI, ECML/PKDD) have been organized on learning ontologies. Most of the work presented there addresses one of the following: a problem of extending an existing ontology WordNet using Web documents [1], using clustering for semi-automatic construction of ontologies from parsed text corpora [2], [16], learning taxonomic, eg., isa, [4], and non-taxonomic, eg., has-Part relations [15], extracting semantic relations from text based on collocations [10], extracting semantic graphs from text for learning summaries [14].

The contribution of this paper to the field is that it presents an novel approach to semi-automatic construction of a very specific type of ontology – topic ontology. The system we developed helps the user at the first out of the three steps from the methodology [19]. Text mining techniques (e.g. clustering) were already proven successful when used at this step (e.g. [2], [16]) and in this paper we presents a very tight integration of them with a manual ontology construction tool. This allows our system to offer support to the user during the whole ontology construction process.

3 Text mining techniques

3.1 Representation of text documents

In order to use the algorithms we will describe later we must first represent text documents as vectors. We use standard *Bag-of-Words* (BOW) approach together with the TFIDF weighting [17]. This representation is often referred to as *vector-space model*. The similarity between two documents is defined as the cosine of the angle between their vector representations – *cosine similarity*. Note that the cosine similarity between two exactly the same documents is 1 and the similarity between two documents that share no common words is 0.

3.2 Latent Semantic Indexing

The language contains many redundant information, since many words share common or similar meaning. For computer this can be difficult to handle without some additional information – background knowledge. Latent Semantic Indexing (LSI), [5], is a technique for extracting this background knowledge from text documents. It uses a technique from linear algebra called Singular Value Decomposition (SVD) and bag-of-words representation of text documents for detecting words with similar meanings. This can also be viewed as extraction of hidden semantic concepts or topics from the text documents.

3.3 K-Means clustering

Clustering is a technique for partitioning data so that each partition (or cluster) contains only points which are similar according to some predefined metric. In the case of text this can be seen as finding groups of similar documents, that is documents which share similar words.

K-Means [12] is an iterative algorithm which partitions the data into k clusters. It has already been successfully used on text documents [18] to cluster a large document corpus based on the document topic and incorporated in an approach for visualizing a large document collection [9].

3.4 Keywords extraction

We used two methods for extracting keywords from a given set of documents: (1) keyword extraction using centroid vectos and (2) keyword extraction using SVM [3]. We used this two methods to generate description for a given topic based on the documents inside the topic.

The first method works by using the centroid vector of the topic (centroid is the sum of all the vectors of the document inside the topic). The main keywords are selected to be the words with the highest weights in the centroid vector.

The second method is based on the idea presented in [3] which uses Support Vector Machine (SVM) binary classifier [13]. Let A be the topic which we want to describe with keywords. We take all the documents from the topics that have A

for a subtopic and mark these documents as negative. We take all the documents from the topic A and mark them as positive. If one document is assigned both negative and positive label we say it is positive. Then we learn a linear SVM classifiers on these documents and classify the centroid of the topic A. Keywords describing the concept A are the words, which's weights in SVM normal vector contribute most when deciding if centroid is positive.

The difference between these two approaches is that the second approach takes into account the context of the topic. Let's say that we have a topic named 'computers'. When deciding, what the keywords for some subtopic A are, the first method would only look at what the most important words within the subtopic A are and words like 'computer' would most probably be found important. However, we already know that A is a subtopic of 'computers' and we are more interested in finding the keywords that separate it from the other documents within the 'computers' topic. The second method does that by taking the documents from all the super-topics of A as a context and learns the most crucial words using SVM.

4 Semi-automatic construction of topic ontology

We view semi-automatic topic ontology construction as a process where the user is taking all the decisions while the computer only gives suggestions for the topics, helps by automatically assigning documents to the topics, helps by suggestiong names for the topics, etc. The suggestions are applied only when the users decides to do so. The computer also helps by visualizing the topic ontology and the documents.

In Figure 1 you can see the main window of the interactive system we developed. The system has three major parts that will be further discussed in following subsections. In the central part of the main window is a visalization of the current topic ontology (Ontology visualization). On the left side of the window is a list of all the topics from this ontology. Here the user can select the topic he wants to edit or further expand into subtopics. Further down is the list of suggested subtopics for the selected topic (Topic suggestion) and the list with all topics that are in relation-ship with the selected topic. At the bottom side of the window is the place where the user can fine-tune the selected topic (Topic management).

4.1 Ontology visualization

While the user is constructing/changing topic ontology, the system visualizes it in real time as a graph with topics as nodes and relations between topics as edges. See Figure 2 for an example of the visualization.

4.2 Topic suggestion

When the user selects a topic, the system automatically suggests what the subtopics of the selected topic could be. This is done by LSI or k-means al-

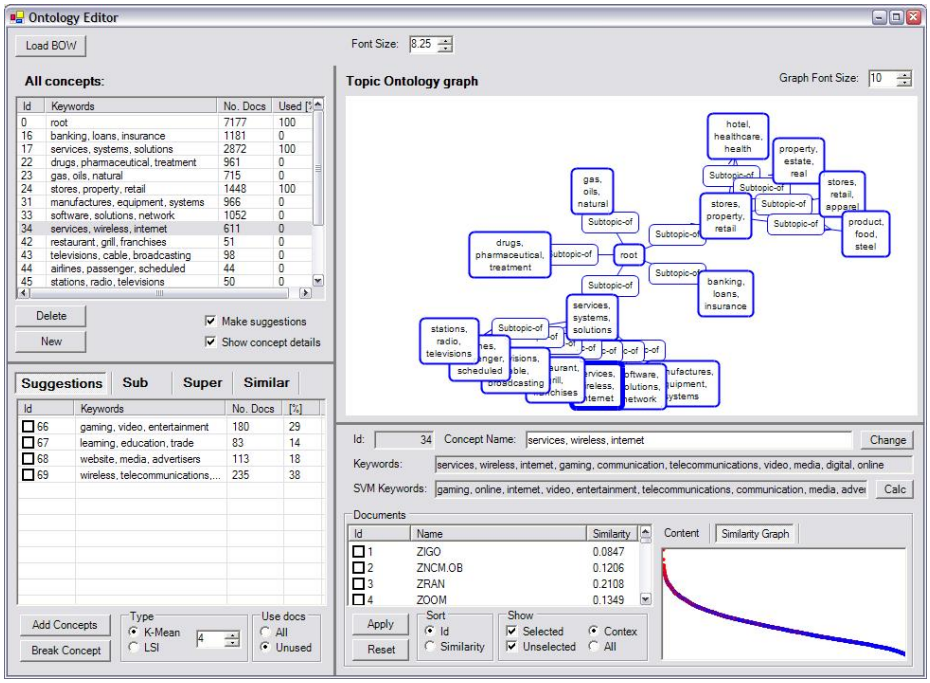


Fig. 1. Screen shot of the interactive system for construction topic ontologies.

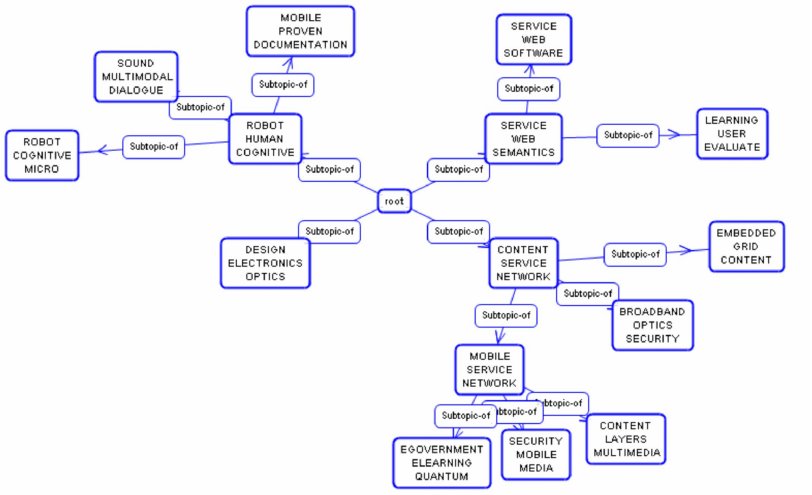


Fig. 2. Example of topic ontology visualization.

gorithms applied only to the documents from the selected topic. The number of suggested topics is supervised by the user. Then, the user selects the subtopics he finds reasonable and the system automatically adds them to the ontology with relation 'subtopic-of' to the selected topic. User can also decide to replace the selected topic with the suggested subtopics. In Figure 3 you can see how is this feature implemented in our system.

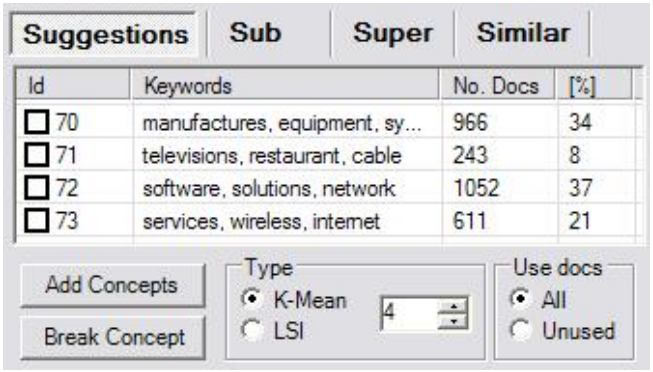


Fig. 3. Example of suggested subtopics.

4.3 Topic management

The user can manually edit each of the topics he added to the topic ontology. He can change which documents are assigned to this topic (one document can belong to more topics), what is the name of the topic and what is the relationship of the topic to other topics. The main relationship is subtopic-of and is automatically added when adding subtopics as described in the previous section. The user can control all the relations between topics by adding, removing, directing and naming the relations.

Here the system can provide help on more levels:

- The system automatically assigns the documents to a topic when it is added to the ontology.
- The system helps by providing the keywords describing the topic using the methods described in Section 3. This can assist user when naming the topic.
- The system computes the cosine similarity between each document from the corpus and the centroid of the topic. This information can assist the user when searching for documents related to the topic. The similarity is shown on the list of documents next to the document name and the graph of similarities is ploted next to the list.
- The system also computes similarities between the selected topic and all the other topics from the ontology. For the similarity measure between two topics it uses either the cosine similarity between their centroid vectors or the intersection between their documents.

See Figure 4 for details on how this is integrated into our system.

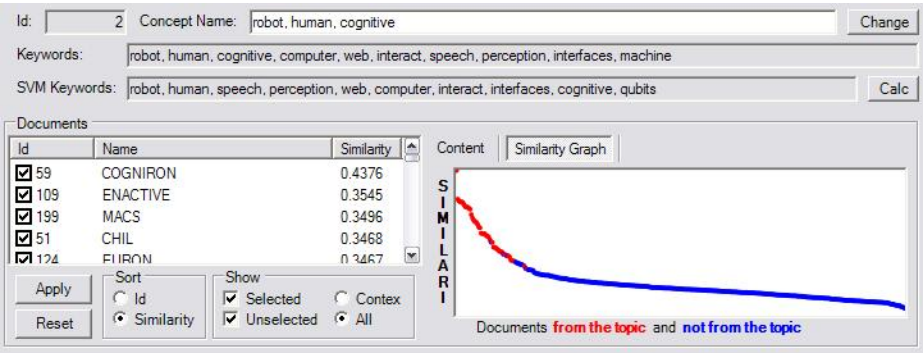


Fig. 4. Topic management.

5 Conclusions

In this paper we presented our approach to the semi-automatic construction of topic ontologies. In the first part of the paper we presented text mining techniques we used: two methods for discovering topics within the corpus, LSI and K-Means clustering, and two methods for extracting keywors. In the second part we showed how we integrated all these methods into an interactive system for constructing topic ontologies.

Since this is work-in-progress there is a large area of possible improvements. The most important next step is to evaluate the proposed system in some practical scenarios and see how it fits the needs of the users and what features are missing or need improvement. Another possible direction is making the whole process more automatic and reduce the need for user interaction. This involves things like calculating the quality of topics suggested by the system, more automated discovery of the optimal number of topics, more support for annotating the documents with the topics, discovering different kinds of relations between topics etc.

We would also like to explore other techniques for concept/topic discovery (for example Probabilistic Latent Semantic Analysis [11] and its derivates) and are considering possible integrations with other tools for ontology building and management.

References

1. AGIRRE, E., ANSA, O., HOVY, E., MARTNEZ, D. (2000): *Enriching very large ontologies using the WWW. In Proceedings of the First Workshop on Ontology Learning OL-2000. The 14th European Conference on Artificial Intelligence ECAI-2000.*

2. BISSON, G., NDELLEC, C., CAAMERO, D. (2000): *Designing clustering methods for ontology building: The MoK workbench. In Proceedings of the First Workshop on Ontology Learning OL-2000. The 14th European Conference on Artificial Intelligence ECAI-2000.*
3. BRANK, J., GROBELNIK, M., MILIC-FRAYLING, N., MLADENIC, D.: Feature selection using support vector machines. Proceedings of the Third International Conference on Data Mining Methods and Databases for Engineering, Finance, and Other Fields, Bologna, Italy, 25–27 September 2002.
4. CIMIANO, P., PIVK, A., SCHMIDT-THIEME, L., STAAB, S. (2004): *Learning Taxonomic Relations from Heterogeneous Evidence. In Proceedings of ECAI 2004 Workshop on Ontology Learning and Population.*
5. DEERWESTER, S., DUMAIS, S., FURNAS, G., LANDUER, T., AND HARSHMAN, R. (1990): *Indexing by Latent Semantic Analysis, , Journal of the American Society of Information Science, vol. 41, no. 6, 391-407*
6. DOUGLAS B. LENAT AND R. V. GUHA (1990): *Building Large Knowledge-Based Systems: Representation and Inference in the CYC Project, Addison-Wesley*
7. FERNNDEZ, L.M. (1999): *Overview Of Methodologies For Building Ontologies. In Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5).*
8. FORTUNA, B., GROBELNIK, M., MLADENIC, D. (2005): *Document atlas.* `http://kt.ijs.si/blazf/software.html`
9. GROBELNIK, M., AND MLADENIC, D. (2002): *Efficient visualization of large text corpora. Proceedings of the Seventh TELRI seminar. Dubrovnik, Croatia*
10. HEYER, G., LUTER, M., QUASTHOFF, U., WITTIG, T., WOLFF, C. (2001): *Learning Relations using Collocations. In Proceedings of IJCAI-2001 Workshop on Ontology Learning.*
11. HOFFMAN, T. (1999): *Probabilistic Latent Semantic Analysis, , Proc. of Uncertainty in Artificial Intelligence, UAI'99*
12. JAIN, MURTY AND FLYNN (1999): *Data Clustering: A Review, ACM Comp. Surv.*
13. T. JOACHIMS (1999): Making large-scale svm learning practical. *In B. Scholkopf, C. Burges, and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning. MIT-Press.*
14. LESKOVEC, J., GROBELNIK, M., MILIC-FRAYLING, N. (2004): *Learning Semantic Graph Mapping for Document Summarization. In Proceedings of ECML/PKDD-2004 Workshop on Knowledge Discovery and Ontologies (KDO-2004).*
15. MAEDCHE, A., STAAB, S. (2001): *Discovering conceptual relations from text. In Proc. of ECAI'2000, pages 321-325.*
16. REINBERGER, M-L., SPYNS, P. (2004): *Discovering Knowledge in Texts for the learning of DOGMA-inspired ontologies. In Proceedings of ECAI 2004 Workshop on Ontology Learning and Population.*
17. SALTON, G. (1991): *Developments in Automatic Text Retrieval, Science, Vol 253, pages 974-979*
18. STEINBACH, M., KARYPIS, G., KUMAR, V. (2000): *A comparison of document clustering techniques. In Proceedings of KDD Workshop on Text Mining, pp. 109110*
19. USCHOLD, M. (1995): *Towards a Methodology for Building Ontologies Workshop on Basic Ontological Issues in Knowledge Sharing, held in conduction with IJCAI-95.*

Conceptual Taxonomy Identification in Medical Documents

Piroska Lendvai

Department of Language and Information Science
Tilburg University
Netherlands

Abstract. We identify taxonomic relations between concepts in medical texts by mining document structure. Taxonomy is induced in the context of a Dutch question answering system, where the reference texts are manually annotated encyclopedia articles. Article sections are assigned a semantic tag if they describe a predefined domain property of the header concept (e.g., `symptoms` or `diagnosis`). When an article has recurring semantic tags in it, the corresponding sections are regarded to stand in taxonomic relationship with the header concept of which they describe subtypes. We propose a supervised learning approach to identify and extract various components of taxonomy from annotated medical encyclopedia texts.

1 Introduction

The question answering (QA) system developed by our group[1] within the Dutch national IMIX project[2] is designed to be able to answer user queries in the medical domain on the basis of a semantically annotated reference text collection. The reference corpus consists of parts of two Dutch medical encyclopedias. In the current version of our module QA is carried out by identifying the medical concept and the semantic topic in the user's question, and matching these to the annotated sentences of the corpus. A small corpus of 67 user questions to the system was collected from naive users by deploying the first version of the QA system. We observed that the concepts in these questions are usually contained in the title of an encyclopedia article (i.e., they are the *header* concept), or in a section title. Therefore, a straightforward strategy is to select candidate documents containing the answer by matching the concept in the question to article headers.

However, we also observed that, perhaps depending on the domain knowledge of the user, the concepts queried may be underspecified with respect to the reference text, as in "What are the symptoms of meningitis?", since the document with the header 'meningitis' has two sections describing symptoms, either of 'viral meningitis' or of 'bacterial meningitis'. In such underspecified

[1] http://ilk.uvt.nl/rolaquad
[2] Interactive Multimodal Information eXtraction

user queries a clarification question has to be generated to the user to specify the correct subconcept, or a warning has to be given about the possible multiple interpretations of 'meningitis'. To enable such intelligent feedback within the QA system, we need to identify documents that contain subtypes of the header concept. The focus of this paper is on developing a machine learning approach to automatically find and extract such conceptual taxonomy in medical documents.

The method exploits manually assigned semantic annotations on various levels of text granularity. The learning algorithm draws on words, as well as concept supercategories and sentence topic types, and classifies whether semantic taxonomy is present between the concepts described by two sections of a document. In our example document, the concepts 'viral meningitis' and 'bacterial meningitis' are described in separate sections that are both annotated with the semantic tags `cause`, `symptom`, and `treatment`. We then label these two sections as taxonomic siblings, because they stand in identical semantic relation(s) with the header concept of the document. Based on labelled examples, a memory-based learner has to identify in our corpus whether two sections of a document are taxonomic siblings or not. We utilise this classification to infer that the concepts 'viral meningitis' and 'bacterial meningitis' are subtypes of the header concept of the given document, 'meningitis', and this knowledge can be added to a domain ontology: *is_a*(viral meningitis,meningitis), *is_a*(bacterial meningitis,meningitis), *taxonomic_siblings*(bacterial meningitis,viral meningitis). The taxonomy can be reused in understanding and clarifying user questions to the QA system.

Our method differs from traditional text-based approaches to generating an ontology, such as using morpho-syntactic pattern matching, or heuristics on unstructured text (see [1] and its references). At the same time, our corpus exhibits less hierarchical structure than scientific or technical texts, utilised e.g. by [2] for inferring domain ontology. Our approach also differs from semantic classification of medical document segments [3], since it discovers taxonomic relations between predefined segments of a document.

In the following section we introduce our reference corpus and its semantic annotation. Next we discuss a number of processes for extracting conceptual taxonomy components from the corpus. In section 4 the results of our first machine learning experiments are reported, and we outline implications for future work. The paper is concluded by a summary.

2 Reference Corpus

Our reference corpus contains parts of the Merck Manual [4] and the Spectrum Medical Encyclopedia [5]. The corpus was automatically segmented on four levels using a tokeniser and exploiting existing XML markup: words, sentences, sections, and documents. The articles were manually annotated based on a protocol for a set of concepts (spanning one or more words), topics (spanning a sentence), and section types of the medical domain. The three tagsets are displayed in Table 1. The corpus contains 3,178 documents. There are 6,582 document (sub)sections; 1,716 documents consist of one section only (54% of the corpus). The average

Document section type	Sentence topic type	Concept supercategory
applications	causes	bodily_function
cause	definition_of	body_part
consequences	diagnoses	disease
contamination	is_a_kind_of	disease_feature
definition	is_property_of	disease_symptom
diagnosis	is_side_effect_of	duration
diseases	is_similar_to	method_of_diagnosis
first_aid	is_symptom_of	microorganism
forms	is_synonym_of	person
incidence	is_transferred_by	person_feature
methods	occurs_in	treatment
prevention	prevents	treatment_feature
side_effects	treats	
symptoms		
treatment		

Table 1. Three semantic annotation tagsets applied to the reference corpus: document section type, sentence topic, concept supercategory.

number of sections present in a document is 2.1. The average document length is 6.3 sentences.

2.1 Semantic annotation types

There are 15 different types of sections annotated in the corpus; this set was defined by examining the main types of domain semantics in the reference text. Sometimes a section cannot be labelled by any of these tags, sometimes more than one tags apply to a section. Subsections are unlabelled. For example, the article titled "Carpal tunnel syndrome" has four sections, respectively labelled as `definition, symptoms, cause`, and `treatment`. Sections are also augmented with their original Dutch encyclopedia section titles (if any), these may correspond to our topic annotations (e.g. in this case 'Symptomen', 'Oorzaak', 'Behandeling'); the first section of multi-section documents usually is a general introductory section, and has no section title.

An example of a more complex structure is the article on "Sterilisation" that has three sections annotated as (1) `definition`, (2) `applications, consequences, definition, method`, and (3) `applications, consequences, definition, method`. The second section's encyclopedia title is 'Bij de man' (i.e., *Of men*), whereas the third section is titled 'Bij de vrouw' (*Of women*). This document is regarded by us to encode some kind of conceptual taxonomy, which is not trivial to discover, since verbally and/or structurally it is only implicitly marked. Our goal is to investigate how such conceptual taxonomy can be identified on the basis of semantic annotations available to us on the concept-, sentence-, and section level.

On the sentence level thirteen semantic topic categories are annotated. It is possible that several different topics are assigned to one sentence. On the concept level twelve semantic supercategories are assigned to domain entities, such as `disease` or `body_part`.

When marked up with the three tag sets, the second sentence of the article on "Hersenvliesontsteking" (i.e. *meningitis*) is annotated as follows:

<SECTION: cause,definition> <TOPIC: definition_of,is_transferred_by,causes> *The disease can be caused by several types of* <CONCEPT: microorganism> *bacteria* </CONCEPT: microorganism> *and* <CONCEPT: microorganism> *viruses* </CONCEPT: microorganism>. </TOPIC: definition_of,is_transferred_by,causes> </SECTION: cause,definition>

3 Extraction of Conceptual Taxonomy Components

To populate an ontology, it is possible to directly extract domain knowledge from the annotated corpus in several ways. Simple methods for extracting domain concepts will be described first.

3.1 Domain concepts

Apart from the manually annotated concepts in the corpus, domain concepts can be generated by listing the article headers; this yields around 3,000 unique medical concepts from our corpus. Such a vocabulary can be further extended when taxonomy is identified in a document, of which the details are described further below (app. 600 more concepts), and by identifying synonyms of the title concept (app. 900 more concepts). The latter technique draws on formal properties of the articles in combination with the manual annotations:

(i) In case the first sentence of an article is tagged with a `definition` topic and begins with "or", the next phrase is identified as a synonym of the title concept: *"or meningitis, an inflammation of the brain membranes."*.

(ii) In case a sentence is tagged with an `is_synonym_of` topic, and contains parentheses, the phrase between parentheses is identified as a synonym of the title concept: *"Bacterial meningitis is caused by bacteria, e.g. by the meningococcus (neisseria meningitidis), the pneumococcus (streptococcus pneumoniae) ..."*, etc.

(iii) In case a header concept is explicitly referred to in the document by the phrase "*See ...*".

3.2 Conceptual taxonomy in encyclopedia articles

In some cases of multi-section documents, we can identify subconcepts of the header concept in the section titles, drawing on the section type annotations. We implemented a procedure that assumes that a section type that is applicable to two or more sections of a document indicates semantic overlap between the

given segments. With this approach we identified 128 documents that exhibit conceptual taxonomy. By manually checking 10% of the identified documents we indeed find taxonomy of domain concepts in these articles: next to subconcepts of the header concept (e.g. malignant vs benevolent tumors of an organ), we can extract taxonomic siblings of treatments (e.g. internal or external application of a medicament), taxonomic siblings of procedures applied to a physiological process (e.g. delivery by cesearean section versus vacuum extractor), and the like.

3.3 Relating domain terms to semantic labels

When a taxonomic relationship is detected in a document, it is possible to mine further knowledge from the relevant sections. The original Dutch section titles of the article can be extracted and stored as terms standing in relation with the given semantic annotation. For example, we can learn from the titles of overlapping section tags that "corticosteroids" have two ways of application (section type `applications`), "internal" and "external". In the article on "saliva" the section type `applications` is assigned both to a section titled as "functies" (*functions*) and to a section titled "samenstelling" (*composition*). Integrating these words in each other's proximity in an ontology would result in improved search terms for a QA module.

3.4 Extracting collocations from a taxonomy

Furthermore, certain lexical items can be extracted from medical documents to create collocations of (medical) terms, nouns, verbs, and n-grams that share the semantic aspect described by the section labels. For example, extracting verbs from the two sections annotated as describing `application` of the header "saliva" yields the list *protect, facilitate, contain, produce, have(_the_function), play(_an_important_role), start(_to_dissolve)*. Such a repository can help identify or create functional paraphrases of the semantic relation in a user query (e.g. "What is the role of saliva during digestion?" and "What functions does saliva have?").

3.5 Clustering documents

Based on the section types an article contains, documents can be clustered together to point out larger groups of medical concepts, and their header concepts can be assigned to predefined domain supercategories. E.g., header concepts from documents with section tags as `definition, symptoms, cause, treatment` are clustered in a pool, the hypernym of which can be generated from the definition sections of the clustered documents, e.g. "illness" (such as meningitis, carpal tunnel syndrome), documents that have section tags as `definition, treatment, side_effects` are clustered in "procedures" (such as sunbath, delivery) , yet others are identified as "diagnostic method" (such as MRI-scan or laparoscopy), sharing the section tags `definition, diagnosis`.

Developing such knowledge from a corpus is useful for our QA module not only in answering medical questions, but also in gaining confidence when analysing the question itself. Erroneously identified concepts or topics in a user query can be filtered out if e.g. a topic is shown to be invalid in some concept cluster, such as queries hypothetically understood as "How do you cure an MRI-scan?".

4 Development of a Machine Learning Approach

All procedures described above require that conceptual taxonomy is automatically discovered in medical documents. We propose to implement this process as a supervised classification task, drawing on our annotated corpus. These exploratory experiments are designed in a bottom-up way: the first step is to automatically learn whether two sections of a document describe taxonomic siblings or not, regardless of the semantic aspect(s) the siblings share. We use a memory-based learner[3] to classify positive and negative instances of taxonomic siblings. Using its default parameters, this algorithm assigns a class to a test instance on the basis of the class of the most similar training example it has seen.

4.1 Feature construction

The learning algorithm draws on a binary vector representing overlap between two sections of a document in terms of an unordered bag-of-words (a lexicon of 5421 bits), as well as on overlap between the set of concept supercategories, and the set of sentence topics (see Table 1). E.g., if both sections have the word "meningitis" in them, the bit representing this word is set to 2; if only one of the sections has the concept `body_part` annotated in it, the bit representing this concept is set to 1; if no sentence in any of the sections is annotated with the topic `is_side_effect_of`, this bit is set to 0, and so on.

4.2 Experimental data and setup

The two encyclopedias we annotated differ from each other in well-structuredness, consistency of section titles, articles' length, etc. To capture the generalisability of our approach, we perform separate experiments on data from the Spectrum Encyclopedia (that consists of highly-structured documents according to a general scheme) and on the Merck Manual (that is built less consistently, its documents bearing more resemblance to descriptive texts where content steers structure). After generating section pairs, we run classification experiments on 697 Spectrum instances (174 positive, 523 negative), and 210 instances from Merck (49 positive, 161 negative).

Since the data are small, we perform a leave-one-out testing method: the whole dataset is used as training data but one datapoint, which is afterwards

[3] TiMBL, release 5.1. `http://ilk.uvt.nl/timbl`

Collection	Features	overall scores			+ class			− class		
		Acc	FmI	FmA	Prec	Rec	F	Prec	Rec	F
Spectrum	BoW	57	58	44	16	17	17	72	70	71
	concepts	75	75	67	51	49	50	83	84	84
	topics	88	88	85	78	76	77	92	93	92
Merck	BoW	67	66	52	27	24	26	78	80	79
	concepts	69	69	56	33	33	33	80	80	80
	topics	79	80	73	55	65	60	89	84	86

Table 2. Classification results of conceptual taxonomy between two sections of a medical document, based on leave-one-out experiments with memory-based learning, varying feature representation.

used for testing. Evaluative metrics are overall accuracy, micro F-score (measured over all instances), macro F-score (measured over both classes), whereas for both classes we calculate precision, recall, and F-score. Our focus of interest is the F-score as measured over the positive class, since this figure characterises how well we are able to identify taxonomic siblings, which then allows for inducing components of an ontology.

4.3 Results and possible extensions

The experimental results are shown in Table 2. On both datasets classification yields the best scores when the two sections are represented by the overlap in annotated sentence topics. Using this information, the learning algorithm successfully detects sections that describe taxonomic siblings with 60 points F-score on Merck data and 77 points F-score on Spectrum data. The type of concept supercategories that can be found in a section also adds important information to guessing a taxonomy, but on a much smaller scale than sentence topics. Overlap of words between the two sections gives the least cues to the learner.

In general, and as expected, the scores are higher on Spectrum data, whereas those on Merck can be seen to represent the real-life situation of inducing taxonomy from relatively free-form encyclopedia articles. Precision and recall on both positive and negative classes are quite balanced, which means that the algorithm is able to treat both classes similarly, despite that positive classes are in minority in both datasets.

We assume that the current results can be employed as baselines in more elaborate experimental implementations of the approach. We plan to extend this setup along the following lines:

- test classifiers other than the memory-based learner
- test classifier parameters different from the defaults
- test features that describe global document properties (document length, amount of sections, etc.)
- employ predicted instead of annotated semantic features

- employ morpho-syntactic features and n-grams
- perform feature selection
- utilise other (medical) encyclopedia data
- merge the ontologies extracted from different datasets.

5 Summary

We have described how components of conceptual taxonomies can be mined from a semantically annotated corpus of Dutch medical encyclopedia texts. This research is motivated by the assumption that such procedures can be generally applied to structured descriptive documents such as encyclopedias or monolingual dictionaries, and when (a portion of) these are annotated on various semantic levels, it is possible to induce domain-specific and/or task-specific ontology components from them.

We are employing the extracted knowledge in a question answering module, creating a taxonomy of domain concepts and their subconcepts, medical supercategories, and lexical representations of these, to use improved search terms and detect underspecification in user queries. Currently the developed techniques are being converted to machine learning procedures implemented into the larger question answering module of our project.

6 Acknowledgements

This research is supported by the Netherlands Organisation for Scientific Research (NWO).

References

1. Cimiano, P., Pivk, A., Schmidt-Thieme, L., Staab, S.: Learning taxonomic relations from heterogeneous sources of evidence. In Buitelaar, P., Magnini, B., Cimiano, P., eds.: Ontology Learning from Text: Methods, Applications, Evaluation. IOS Verlag (2005)
2. Makagonov, P., Figueroa, A., Sboychakov, K., Gelbukh, A.: Learning a domain ontology from hierarchically structured texts. In: Proc. of ICML workshop on Learning and Extending Lexical Ontologies by using Machine Learning Methods. (2005) 50–57
3. Cho, P., Taira, R., Kangarloo, H.: Automatic segmentation of medical reports. In: Proc. of AMIA Symposium. (2003) 155–159
4. Berkow, R., ed.: Merck Manual Medisch handboek. Bohn Stafleu Van Loghum (2000)
5. Spectrum: Winkler Prins Medische Encyclopedie. Spectrum (2003)

Evaluation of Ontology Enhancement Tools ⋆

Myra Spiliopoulou[a], Markus Schaal[b] ⋆⋆, Roland M. Müller[a], and Marko Brunzel[a]

[1] [a] Otto-von-Guericke-Universität Magdeburg
[2] [b] Bilkent University, Ankara

Abstract. Mining algorithms can enhance the task of ontology establishment but methods are needed to assess the quality of their findings. Ontology establishment is a long-term interactive process, so it is important to evaluate the contribution of a mining tool at an early phase of this process so that only appropriate tools are used in later phases. We propose a method for the evaluation of such tools on their impact on ontology enhancement. We model impact as quality perceived by the expert and as statistical quality computed by an objective function and we provide a mechanism that juxtaposes the two forms of quality. We have applied our method on an ontology enhancement tool and gained some interesting insights on the interplay between perceived impact and statistical measures.

1 Introduction

The manual establishment of ontologies is an intriguing and resource-consuming task. Efforts are made to enhance this process by unsupervised learning methods. However, as pointed out in [10], the semantic richness and diversity of corpora does not lend itself to full automation, so that the involvement of a domain expert becomes necessary. Hence, unsupervised tools undertake the role of providing useful suggestions, whereupon the quality of their contributions must be evaluated. Since ontology enhancement is a long-term process involving multiple corpora and possibly multiple iterations over the same corpus, this evaluation should be done at an early step, so that only appropriate tools are considered in later steps. In this study, we propose a method for the early evaluation of clustering tools that suggest concepts for ontology enhancement.

Our method has two aspects: First, it evaluates the *impact* of the tool's suggestions as *perceived* by the domain expert. Second, it juxtaposes the *objective quality* of these suggestions to the perceived impact. While the objective quality refers to the statistical properties of the discovered patterns, such as the confidence of a rule or the homogeneity of a cluster, the impact is reflected in the ultimate decision of the expert to include the suggested pattern in the ontology or not. The juxtaposition of the objective, tool-internal notion of quality to the quality perceived by the expert indicates whether the tool and its quality measures will be helpful in further steps of the ontology establishment process.

In the next section, we discuss related work on the evaluation of unsupervised learning tools. In section 3 we describe our method for impact evaluation by the domain expert, followed by the method juxtaposing impact and statistical quality. In section 4, we briefly present the tool we have used as experimentation example. Section 5 describes our experiments and acquired insights. The last section concludes our study.

⋆ Work partially funded under the EU Contract IST-2001-39023 Parmenides.
⋆⋆ Work done while with the Otto-von-Guericke-Universität Magdeburg

2 Related Work

Ontology learning tools as proposed in [1, 2, 4, 6, 9, 8, 13] serve different purposes. Many of them propose objects (concepts and relationships) that are found to be supported by a document collection relevant to the application at hand. We concentrate on tools that enhance an existing ontology by proposing (a) new concepts to be inserted in it and (b) relationships among existing concepts.

Usually, an ontology enhancement tool has an inherent quality assessment mechanism that rejects patterns according to some scheme. For tools based on association rules' discovery, quality assessment is often based on interestingness and unexpectedness, while cluster quality is often based on homogeneity or compactness. A rich collection of criteria for the statistical evaluation of unsupervised learners has appeared in [15]. It contains valuable criteria for the assesment of cluster quality, many of them based on indexes of cluster homogeneity. More oriented towards the needs of text clustering are the criteria considered in [14], in which a correlation between some cluster homogeneity indexes and the F-measure is identified when experimenting upon a gold standard. However, application-specific ontology learning cannot rely on gold standards developed for different applications. Moreover, cluster homogeneity does not guarantee or imply that the cluster labels will also be interesting to the domain expert.

Evaluation from the viewpoint of ontology learning is more challenging. Holsapple and Joshi proposed an evaluation method for collaborative manual ontology engineering, in which each suggestion made by one expert is evaluated by at least another expert [7]. Hence, good suggestions are those that enjoy the approval of multiple experts. While this is reasonable for ontology engineering among human experts, it cannot be transferred to non-human experts: Agreement among several ontology learners does not necessarily imply that human experts will find their suggestions useful, since ontology learners are based more on statistics than on background knowledge and expert insight.

The ECAI 2004 workshop on "Ontology Learning and Population" concentrated on the subject of "Evaluation of Text-based Methods in the Semantic Web and Knowledge Discovery Life Cycle" [1]. Faatz and Steinmetz proposed an elegant formalization of "ontology enrichment", followed by a method for automated evaluation on the basis of precision and recall [3], i.e. with respect to gold standards. The selection of those measures is in accordance with the task of evaluation *for algorithmic tuning*: The authors state that "only automatic evaluations of ontology enrichment meet the requirements of algorithmic tuning" and that "the automatization has to be aware of the task specific semantic direction, to which an ontology should evolve" [3]. In our study, we pursue a different goal: We want to assist an expert in deciding on the appropriateness of the tool rather than tune any tool. Moreover, we deliver a procedure that decides whether algorithmic tuning should be made or rather avoided as incompatible to the preferences/intuition of the expert.

Porzel and Malaka consider task-oriented evaluation of ontologies [12]. The process creating an ontology is not specified explicitly, but (semi-)automated processes seem to be permissible; a tool could be evaluated on the quality of the ontology it produces. The authors consider evaluation only with respect to a predefined task, since

[1] http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005

ontologies are indeed built to serve specific tasks. Their evaluation method is based on error-rates, namely superfluous, ambiguous or missing concepts with respect to the task [12]. For our objective of appropriateness evaluation for tools, this approach has some shortcomings. First, it evaluates whole ontologies, while we are interested in the step-wise enhancement of a preliminary ontology. Second, evaluation on the basis of error rates requires a gold standard tailored to the anticipated task. The establishment of such a standard is quite counterintuitive from the viewpoint of a domain expert that needs a tool to enhance an ontology with concepts she does not know in advance.

Navigli et al proposed an evaluation method for the OntoLearn system, encompassing a quantitative evaluation towards specific corpora and a qualitative evaluation by multiple domain experts [11]: Quantitative evaluation for the term extraction algorithm, the ontology learning algorithm and the semantic annotation algorithm was performed on predefined corpora which served as gold standards. While this type of evaluation allows for conclusions about the robustness of one tool or the relative performance of multiple tools, it does not allow for generalizations on the usefulness of a given tool to a given expert for the enhancement of a given ontology from a given document collection.

The qualitative evaluation proposed in [11] was based on a questionnaire, in which experts assessed the quality of the definitions of the concepts discovered by OntoLearn: The complex concepts found by the OntoLearn rules were combined with concept definitions from WordNet. The experts were then asked to rate the glosses thus generated as unacceptable, helpful or fully acceptable. This is closer to our one-expert evaluation. However, we do not consider concept definitions, because (a) an appropriate definition provider may or may not be available – the WordNet is not appropriate for specialized domains, (b) the interpretation of a complex concept is left to the expert and (c) a small or medium enterprise intending to enhance an ontology is more likely to dedicate one domain expert to this task rather than 10 or 25 experts. So, the approach is not applicable for *providing assistance* to *one* expert. Further, the appropriateness of the selected corpus was taken for granted; in our approach, this assumption is being put to test.

A method for the generation and evaluation of suggestions towards an ontology user is proposed in [5]. The authors propose a recommendation engine that explores the activities of multiple users, who expand their personal ontologies from a shared basic ontology and suggest metrics for the evaluation of the engine's suggestions. This approach is appropriate when ontologies are built collaboratively by people, since the actions of one person may be helpful to others. However, the metrics do not apply for the actions (suggestions) of one tool towards one domain expert.

3 A Posteriori Impact Evaluation for Ontology Enhancement

Our method evaluates the impact of text miners on the task of ontology enhancement. A text miner processes a text collection and suggests semantics that expand the ontology. These may be terms like "hurricane" or "hurricane warning", term groups that form a new concept or a relation among concepts, e.g. "hurricane warning area", or named relations like "expected within". We refer to them as *"concept constellations"* and focus on the evaluation of the process discovering them. We focus on tools for text clustering and labeling, but our method can be easily extended for association rules' discovery.

3.1 Objectives

We observe ontology enhancement as an iterative process on one or more document collections. Its initial input is a preliminary ontology to be enhanced with help of each application-specific collection. Its final output should be an enriched ontology that is "complete towards the collection", in the sense that the collection cannot contribute new concept constellations to it. Our evaluation method is intended for the first step of this process and should answer the following questions:

- Is the tool appropriate for the enhancement of *this* ontology – on *this collection*?
- Is the collection appropriate for the enhancement of the ontology – with this tool?
- Are the quality evaluation functions of the tool aligned to the expert's demands?

The motivation of the first question is that a tool may perform well for one collection and poorly for another. A collection can itself be inappropriate for the enhancement of the specific ontology, e.g. if for each feature space the tool builds, the expert rejects all the features. We deal with the first two questions hereafter and with the last one in 3.4.

3.2 Perceived Quality as Relevance + Appropriateness

We evaluate the tool's impact on ontology enhancement as *perceived* by the ontology expert. We use two criteria, "relevance to the application domain" $R(D)$ and "appropriateness for the ontology O" $A(O, D)$, where D stands for the collection *as representative of the application Domain*.

Relevance to the Application Domain. The ontology enhancement is assumed to take place in the context of an application domain and that the collection is representative of that domain. For this criterion, the domain expert is asked to characterize each suggestion (concept constellation) made by the tool as relevant or irrelevant to that domain, *independently of whether she considers the suggestion as appropriate for the ontology*.

The term "relevance" is known to be very subjective. However, the intention of this criterion is not to assess the relevance of the individual suggestions but rather the appropriateness of the tool and of the collection for the application domain. In particular, consider the following excerpt from the National Hurricane Center at `www.noaa.com`:

```
A HURRICANE OR TROPICAL STORM WARNING MEANS THAT HURRICANE OR
TROPICAL STORM CONDITIONS ... RESPECTIVELY ... ARE EXPECTED WITHIN
THE WARNING AREA WITHIN THE NEXT 24 HOURS. PREPARATIONS TO PROTECT
LIFE AND PROPERTY SHOULD BE RUSHED TO COMPLETION IN THE HURRICANE
WARNING AREA.
```

For the application area of extreme weather warnings, a tool applied on the text collection might suggest the following concepts / constellations, listed here in alphabetical order: (I) "storm, tropical, warning", "area, hurricane, warning", "preparations, protect", (II) "hurricane", "storm", (III) "are, expected", "area". Note that we do not check whether the tool can assess that e.g. "hurricane warning area" is one or two concepts.

- Suggestions of type I are relevant. If most suggestions are of this type, then the tool is appropriate for the collection.

– Suggestions of type III are irrelevant and indicate that the tool cannot find relevant concept constellations upon this collection. If most suggestions are of this type, it should be checked whether the collection itself is appropriate. If yes, then the tool is not appropriate for it.
– Type II suggestions are more challenging. An expert may reject the suggestion "hurricane" as uninformative for an application domain on hurricanes. However, with respect to our criterion, such suggestions should be marked as relevant: Informativeness and appropriateness for the ontology are addressed by our next criterion.

Appropriateness for the Ontology. The Appropriateness criterion $A(O, D)$ refers to the expansion of ontology O for the application domain D. It builds upon the relevance criterion $R(D)$: only relevant concept constellations are considered. For a relevant concept constellation $Y = Y_1, \ldots, Y_m$, the following cases may occur:

– Y is already in the ontology. Then it should be rejected as inappropriate.
– Y contains some concepts that are appropriate for the ontology, either as individual concepts or as a group. Then Y should be accepted; each appropriate concept/group should be named.
– Y contains no concept that is appropriate for the ontology. It should be rejected.

According to this scheme, a concept constellation may contribute one or more concepts to the ontology. Hence, $A(O, D)$ delivers two lists of results: $A(O, D) = \{S, S_+\}$, where $S \subseteq R(D)$ is the set of accepted concept constellations and $S_+ \in \mathcal{P}(S)$ is the set of concept groups appropriate for the ontology.

We use the result $A(O, D).S$ to assess the appropriateness of the tool for further iterations in the ontology enhancement process. The result $A(O, D).S_+$ is used in 3.4, where we juxtapose the quality criteria of the tool to the impact perceived by the expert.

3.3 Combining Relevance and Appropriateness Ratings

Let $T(D)$ be the set of concept constellations suggested by the tool T for the application domain. We combine the results on relevance $R(D) \subseteq T(D)$ and appropriateness for the ontology $A(O, D).S$ to figure out whether the tool T should be further used for the enhancement of the ontology on domain D, whereupon we consider the collection already analyzed as representative for domain D. The following cases may occur:

– The ratio $\frac{|R(D)|}{|T(D)|}$ is close to zero.
Then, the tool is not appropriate for this collection and thus for the domain.
– The ratio $\frac{|R(D)|}{|T(D)|}$ is closer to one and the ratio $\frac{|A(O,D).S|}{|R(D)|}$ is close to zero.
Then, the tool is capable of analyzing documents in the application domain but the collection does not deliver informative concept constellations for the ontology. This may be due to the tool or to the relationship between ontology and collection. To exclude the latter case, the domain expert should again verify the appropriateness of this collection *for ontology enhancement*: If all concepts in the collection are already in the ontology, the collection is still relevant but cannot enrich the ontology any more. Hence, the tool should be tested upon another representative collection.

– Both ratios are closer to one than to zero.
  Then, the tool is able to contribute to ontology enhancement for this collection and is thus appropriate for the application domain.

By this procedure, we can assess whether a given tool should be further used for the gradual enhancement of the ontology. It remains to be tested whether the quality criteria used by the tool reflect the quality perceived by the domain expert. To this purpose, we juxtapose the evaluation of the expert to the internal quality evaluation of the tool.

3.4 Juxtaposition of Statistical and Perceived Quality

Each (text clustering) tool has some internal or external criterion for the rejection of potentially poor patterns and the maintenance, respectively further exploitation, of good patterns. The results of any clustering algorithm encompass both good and less good clusters, whereby goodness is often measured in terms of compactness, homogeneity, informativeness etc [14, 15]. We name such criteria "statistical quality criteria".

Towards our objective of ontology enhancement, we say that a statistical quality criterion $SQ()$ *"is aligned to the perceived quality"* when the likelihood that the domain expert considers a concept group as appropriate for the ontology increases (resp. decreases) with the statistical quality of the cluster with respect to that criterion.

As basis for the statistical quality, let $SQ()$ be a statistical quality criterion that assigns to each cluster generated by T a value. Without loss of generality, we assume that the range of these values is $[0, 1]$ and that 1 is the best value. As basis for the perceived quality, we consider the concept groups characterized by the domain expert as appropriate for the ontology, i.e. the set $A(O, D).S_+$ defined in 3.2. Since an element of this set may appear in more than one concept constellations, it can be supported by one or more clusters generated by the tool and these clusters may be of different statistical quality. To juxtapose statistical and perceived quality, we perform the following steps:

1. We partition the valuerange of $SQ()$ into k intervals.
2. We assign to each interval $I[j], J = 1 \ldots k$ the concept groups from $A(O, D).S_+$ supported by clusters with quality in this interval. They form the set $expertApproved(j)$. The algorithm to this purpose (Table 1) selects for each concept group the best quality cluster supporting it.
3. We assign to each interval $I[j]$ the concept constellations in $T(D) \setminus A(O, D).S$. These are the concept constellations rejected as irrelevant or inappropriate for the ontology and form the set $expertRejected(j)$. Differently from the concept groups which may be supported by several clusters, a concept constellation corresponds to exactly one cluster, so the assignment is trivial.

The result of these steps is a pair of histograms across k intervals. The histogram hA contains the numbers of expert-accepted clusters, while hR contains the numbers of expert-rejected clusters. We consider the following cases:

– Both histograms are unimodal, hA is shifted towards the best quality value for $SQ()$, while hR is shifted towards the worst value.
  This is the best case: The likelihood that a cluster contributes to ontology enhancement increases with its quality and vice versa. $SQ()$ is aligned to perceived quality.

```
1  For each concept group x in A(O,D).S+
2     maxSQ(x) = 0
3     For each cluster C in T(D) that supports x
4        if maxSQ(x) less than SQ(C)
5           then maxSQ(x) := SQ(C)
6     Endfor
7  Endfor
8  Partition the range of SQ() into k intervals I[1],...,I[k]
9  Assign each x in A(O,D).S+ to the interval containing maxSQ(x)
```

Table 1. Associating each concept group to the best quality cluster

– Both histograms are unimodal, hR is shifted towards the best value and hA is shifted towards the worst value.
  This is the second best case. The statistical quality criterion is *consistently* counterproductive. One might reasonably argue that this $SQ()$ is a poor criterion, but it is also true that $1 - SQ()$ is aligned to the perceived quality and is thus very useful.
– The two histograms have the same shape and are shifted in the same direction.
  Then the likelihood of having a good cluster accepted or rejected by the expert is the same as for a bad cluster. Thus, $SQ()$ is misaligned to the perceived quality.
– No pattern can be recognized. Then $SQ()$ is misaligned to the perceived quality.

By this juxtaposition we can assess whether (and which among) the statistical quality criteria used by the tool are aligned to the implicit perceived quality function of the domain expert. If some criteria are aligned, they should take priority over misaligned ones in subsequent ontology enhancement steps. Even if all criteria are misaligned, the tool can still be used. However, it should then deliver to the domain expert the poor quality clusters as well, since she may find useful information in them.

4 An Example Tool and its Quality Evaluation Criteria

As a proof of concept, we have applied our evaluation methodology upon the tool "RELFIN Learner" [13]. We describe RELFIN and its internal quality evaluation criteria below, mostly based on [13]. We stress that RELFIN is only an example: Our method can be applied on arbitrary tools that suggest concepts for ontology enhancement. Obviously, the juxtaposition to a tool's statistical quality is only feasible if the tool reports its quality assessment values as required in 3.4.

RELFIN is a text clustering algorithm using Bisecting-K-means as its clustering core and a mechanism for cluster evaluation and labeling. RELFIN discovers new concepts as single terms or groups of terms characterizing a cluster of text units. These concepts, resp. concept constellations can be used to expand the existing ontology, to semantically tag the corresponding text units in the documents or to do both. RELFIN can take as input both concepts from an initial, rudimentary ontology and with additional terms it extracts automatically from the collection. Accordingly, its suggestions

are new concepts consisting of terms in the collection and constellations consisting of terms from either the ontology or the collection. The labels / concept constellations suggested by RELFIN should be appropriate as semantic markup on the text fragments. This is reflected in the quality criteria of RELFIN.

4.1 Definitions

A *text unit* is an arbitrary text fragment extracted by a linguistic tool, e.g. by a sentence-splitter; it is usually a paragraph or a sentence. Text units are composed of terms. For our purposes, a *text collection* $\mathcal{A}$ is a set of text units.

A term is a textual representation of a *concept*. A *feature space* $\mathcal{F}$ consists of concepts from the existing ontology, terms extracted from the collection by some statistical method or both. We assume a feature space with d dimensions and a *vectorization* $\mathcal{X}$ in which each text unit i is represented as vector of TFxIDF weights $x_i = (x_{i1}, \ldots, x_{id})$. Obviously, concepts of the ontology that do not appear in the collection are ignored.

Given is a *clustering scheme* or *clusterer* $\mathcal{C}$. For a cluster $C \in \mathcal{C}$, we compute the in-cluster-support of each feature $f \in \mathcal{F}$ as

$$ics(f, C) = \frac{|\{x \in C | x_f \neq 0\}|}{|C|} \tag{1}$$

Definition 1 (Cluster Label). *Let $C \in \mathcal{C}$ be a cluster over the text collection $\mathcal{A}$ for the feature space $\mathcal{F}$. The* label *of C $label(C)$ is the set of features $\{f \in \mathcal{F} | ics(f, C) \geq \tau_{ics}\}$ for some threshold τ_{ics}.*

A feature satisfying the threshold constraint for a cluster C is a *frequent feature* for C.

4.2 Quality Measures

A label might be specified for any cluster. To restrict labeling to good clusters only, we use one criterion on cluster compactness and one on feature support inside clusters.

Definition 2 (Average distance from centroid). *Let $C \in \mathcal{C}$ be a cluster over the text collection $\mathcal{A}$ for the feature space $\mathcal{F}$ and let $d()$ be the distance function for cluster separation. The average intra-cluster distance from the centroid is defined as $avgc(C) = \frac{\sum_{x \in C} d(x, centroid(C))}{|C|}$, whereupon lower values are better.*

Definition 3 (Residue). *Let $C \in \mathcal{C}$ be a cluster and let τ_{ics} be the in-cluster support threshold for the cluster label. Then, the "residue" of C is the relative in-cluster support for infrequent features:*

$$residue(C, \tau_{ics}) = \frac{\sum_{f \in \mathcal{F} \setminus label(C)} ics(f, C)}{\sum_{f \in \mathcal{F}} ics(f, C)} \tag{2}$$

The residue criterion serves the goal of using cluster labels for semantic markup. Consider text units that support the features X and Y and text units that support Y and Z. If the algorithm assigns them to the same cluster, then both pairs of features can be

frequent, depending on the threshold τ_{ics}. A concept group "X,Y,Z" may well be of interest for ontology enhancement, but it is less appropriate as semantic tag. We allow for low τ_{ics} values, so that such constellations can be generated. At the same time, the residue criterion favours clusters dominated by a few frequent features shared by most cluster members, while all other features are very rare (values close to zero are best).

5 Experiments

We performed an experiment on ontology enhancement involving a domain expert who used the RELFIN Learner for the enhancement of an existing ontology. The expert's goal was to assess usability of the tool. The complete usability test is beyond the scope of this study, so we concentrate only on the impact assessment criteria used in the test. The juxtaposition to the statistical criteria of the tool was not part of the usability test.

5.1 The Case Study for Ontology Enhancement

Our methodology expects a well-defined application domain. This was guaranteed by a predefined case study with a given initial ontology on biotechnology watch and two domain-relevant collections of business news documents. We used a subcollection of BZWire news (from 1.1.2004 to 15.3.2004), containing 1554 documents. The vectorization process resulted in 11,136 text fragments.

The feature space consisted of 70 concepts from the initial ontology and 230 terms extracted from the collection. These terms were derived automatically as being more frequent for the collection than for a reference general purpose corpus. The target number of clusters was set to 60 and the in-cluster-support threshold for cluster labeling τ_{ics} was set to 0.2. Setting τ_{ics} to such a rather low value has turned to be helpful for our observations, because high values would reduce the set of suggestions considerably.

5.2 Evaluation on Relevance and Appropriateness

RELFIN delivered 60 clusters of varying quality according to the tool's internal criteria. For the impact assessment by the domain expert, though, these criteria were switched off, so that all cluster labels subject to $\tau_{ics} = 0.2$ were shown to the domain expert. This implies that RELFIN suggested the labels of all 60 clusters, so that $|T(D)| = 60$.

The domain expert was asked to assess the relevance of each cluster label, i.e. constellation of frequent features. A label was relevant if it contained at least one relevant feature. The appropriateness of the features in relevant cluster labels was assessed next: The domain expert was asked whether NONE, ALL or SOME of the concepts in the relevant label were also appropriate. The answers were:

- *Relevance to the case study:* YES: 43, NO: 17 $|R(D)| = 43$
- *Appropriateness for the ontology:* NONE: 2, ALL: 4, SOME: 37 $|A(O, D).S| = 41$

We combined these values as described in 3.3. To compute $A(O, D).S_+$, we enumerated the concept groups in the labels characterized as SOME, using the following rules:

1. The expert saw a label with several concepts and named n concept groups that he considered appropriate for the ontology. Then, we counted n appropriate objects.
2. The expert found an appropriate concept or concept group and marked it in *all* cluster labels containing it. Then, we counted the appropriate object only once.
3. The domain expert saw a label "A,B,C,...", and wrote that "A,B" should be added to the ontology. Then, we counted one appropriate object only, even if the terms "A" and "B" did not belong to the ontology.
4. The expert saw a label of many concepts and marked them "ALL" as appropriate. This case occured 4 times. For three labels, we counted one appropriate object only, independently of the number of new concepts and possible combinations among them. For the 4th label, we counted two appropriate objects: the label as a whole and one specific term X. X belongs to a well-defined set of terms and the expert had encountered and accepted three further members of this set when evaluating other clusters. So we added this term, too.

In Table 2 we show the relevance and appropriateness ratios according to those rules. These ratios allow for an assessment (positive in this case) of the tool's appropriateness for further iterations. In the last rows, we have computed the average number of appropriate concept groups, as contributed by the RELFIN clusters. The last ratio is peculiar to RELFIN, which can exploit both concepts from the ontology and terms from the collection. The ratio says that 87% of the approved concept groups were not in the ontology. The remaining 23% are combinations of concepts from the ontology.

Tool suggestions	$\lvert T(D)\rvert$	60
Relevance ratio	$\frac{\lvert R(D)\rvert}{\lvert T(D)\rvert}$	$43/60 \approx 0.72$
Appropriateness ratio	$\frac{\lvert A(O,D).S\rvert}{\lvert R(D)\rvert}$	$41/43 \approx 0.95$
Avg contribution of concept groups per relevant cluster		$62/43 \approx 1.44$
Avg contribution of concept groups per cluster		$62/60 \approx 1.03$
Contribution of the collection to the ontology		$54/62 \approx 0.87$

Table 2. Relevance and appropriateness ratios

5.3 Impact versus Statistical Quality

For the juxtaposition of the impact evaluation with the statistical quality criteria of RELFIN, we used the approach described in 3.4. Both criteria used by RELFIN range in the interval $[0, 1]$; 1 is the worst value and 0 is the best one. We have adjusted the generic procedure accordingly for the experiment.

Table 3 shows that the criterion "average distance to the centroid" $avgc()$ is aligned to the expert's evaluation: For the $avgc()$, most values of the hA histogram (expertApproved clusters) are in $[0.3, 0.5)$; a steep decrease occurs afterwards. In the hR (expertRejected clusters), most values are in $[0.5, 1)$.

The $residue()$ criterion is misaligned. Both hA and hR have a shift towards 1 and the largest modus is in the interval $[0.6, 1]$. For hR, there is a smaller modus in the

	Avg Distance to centroid					
	[0,0.2)	[0.2,0.3)	[0.3,0.4)	[0.4,0.5)	[0.5,1]	
Approved concept groups	2	7	19	27	6	
expertApproved clusters	2	5	12	17	7	
expertRejected clusters	1	1	1	4	10	
	Residue					
	[0,0.2)	[0.2,0.3)	[0.3,0.4)	[0.4,0.5)	[0.5,0.6)	[0.6,1]
Approved concept groups	0	2	6	16	12	25
expertApproved clusters	0	2	4	9	9	19
expertRejected clusters	1	3	4	0	3	6

Table 3. Quality values for approved vs rejected clusters

interval $[0.3, 0.4)$, indicating that clusters characterized as good by this criterion are likely to be rejected. As explained in 3.4, this could be the second-best case. However, the existence of two comparable modi in hR indicates that the criterion is misaligned.

One explanation of the misalignment of the residue is that the labels of clusters with higher residue contain more concepts. When the human expert identified appropriate concept groups for the ontology, he had more candidates to choose from. Those concept groups are not appropriate as semantic tags but this does not affect their appropriateness for the ontology. We consider this as indicatory for impact assessment: If a concept (group) appeals to the domain expert, i.e. is informative with respect to her background knowledge, she will approve it independently of its statistical support.

6 Conclusions

We have proposed a methodology that evaluates the appropriateness of text clustering tools for ontology enhancement on the basis of their suggestions to the domain expert. Our approach is intended as an instrument to help the domain expert decide at the beginning of the ontology enhancement process whether the tool is appropriate for further steps of this process. To this purpose, we combine subjective impact assessment with a more objective relevance test and we finally check whether the statistical evaluation instruments used by the tool are aligned to the subjective preferences of the expert. We have performed a first test of our methodology for a text clustering tool on the enhancement of the ontology of a real case study and we gained some rather interesting insights on the interplay of statistical "goodness" and subjective "appropriateness".

The juxtaposition of statistical quality and impact assessment might be observed as a classification task, where statistical criteria serve as predictors of impact. We intend to investigate this potential. We further plan to enhance the impact assessment with more elaborate criteria. Moreover, we want to evaluate further tools with our methodology: This implies conducting an experiment in which the expert works with multiple tools on the same corpus and the same basic ontology.

Acknowledgement. We would like to thank the domain expert Dr. Andreas Persidis of the company BIOVISTA for the impact evaluation and for many insightful comments on the expectations towards interactive tools used in ontology enhancement.

References

1. Philipp Cimiano, Steffen Staab, and Julien Tane. Automatic acquisition of taxonomies from text: Fca meets nlp. In *Proc. of the ECML/PKDD Workshop on Adaptive Text Extraction and Mining*, pages 10–17, Cavtat, Croatia, Sept. 2003.
2. Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R. Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, and Jason Y. Zien. SemTag and seeker: Bootstrapping the semantic web via automated semantic annotation. In *Proc. of the 12th Int. World Wide Web Conf.*, pages 178–186, Budapest, Hungary, 2003. ACM Press.
3. Andreas Faatz and Ralf Steinmetz. Precision and recall for ontology enrichment. In *Proc. of ECAI-2004 Workshop on Ontology Learning and Population*, Valencia, Spain, Aug. 2004. http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.
4. David Faure and Claire Nédellec. Knowledge acquisition of predicate argument structures from technical texts using machine learning: The system ASIUM. In Dieter Fensel and Rudi Studer, editors, *Proc. of 11th European Workshop on Knowledge Acquisition, Modeling and Management (EKAW'99)*, volume LNAI 1621, pages 329–334, Dagstuhl, Germany, May 1999. Springer-Verlag, Heidelberg.
5. Peter Haase, Andreas Hotho, Lars Schmidt-Thieme, and York Sure. Collaborative and usage-driven evolution of personal ontologies. In *Proc. of European Conference on the Semantic Web (ESWC 2005)*, LNCS 3532, pages 486–499. Springer Verlag Berlin Heidelberg, May/June 2005.
6. Siegfried Handschuh, Steffen Staab, and F. Ciravegna. S-CREAM – Semi-automatic CREation of metadata. In *Proc. of the European Conf. on Knowledge Acquisition and Management*, 2002.
7. Clyde Holsapple and K.D. Joshi. A collaborative approach to ontology design. *Communications of ACM*, 45(2):42–47, 2005.
8. Andreas Hotho, Steffen Staab, and Gerd Stumme. Explaining text clustering results using semantic structures. In *Proc. of ECML/PKDD 2003*, LNAI 2838, pages 217–228, Cavtat-Dubrovnik, Croatia, Sept. 2003. Springer Verlag.
9. Jianming Li, Zhang Lei, and Yong Yu. Learning to generate semantic annotation for domain specific sentences. In *Proc. of the 'Knowledge Markup and Semantic Annotation" Workshop of the K-CAP 2001 Conference*, 2001.
10. Alexander Maedche and Steffen Staab. Semi-automatic engineering of ontologies from text. In *Proc. of 12th Int. Conf. on Software and Knowledge Engineering*, Chicago, IL, 2000.
11. Roberto Navigli, Paola Velardi, Alessandro Cucchiarelli, and Francesca Neri. Quantitative and qualitative evaluation of the ontolearn ontology learning system. In *Proc. of ECAI-2004 Workshop on Ontology Learning and Population*, Valencia, Spain, Aug. 2004. http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.
12. Robert Porzel and Rainer Malaka. A task-based approach for ontology evaluation. In *Proc. of ECAI-2004 Workshop on Ontology Learning and Population*, Valencia, Spain, Aug. 2004. http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.
13. Markus Schaal, Roland Mueller, Marko Brunzel, and Myra Spiliopoulou. RELFIN - topic discovery for ontology enhancement and annotation. In *Proc. of European Conference on the Semantic Web (ESWC 2005)*, LNCS 3532, pages 608–622, Heraklion, Greece, May/June 2005. Springer Verlag Berlin Heidelberg.
14. Benno Stein, Sven Meyer zu Eissen, and Frank Wißbrock. On Cluster Validity and the Information Need of Users. In M.H. Hanza, editor, *3rd IASTED Int. Conference on Artificial Intelligence and Applications (AIA03)*, pages 216–221, Benalmadena, Spain, September 2003. ACTA Press.
15. Michalis Vazirgiannis, Maria Halkidi, and Dimitrios Gunopoulos. *Uncertainty Handling and Quality Assessment in Data Mining*. Springer, 2003.

Towards the Reduction of Spatial Joins for Knowledge Discovery in Geographic Databases Using Geo-Ontologies and Spatial Integrity Constraints

Vania Bogorny, Paulo Martins Engel, Luis Otavio Campos Alvares

Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil
{vbogorny, engel, alvares}@inf.ufrgs.br

Abstract. Spatial join is the most expensive operation in geographic databases, but essentially important to compute spatial relationships intrinsic to geographic data. In account of spatial relationships real world entities may affect the behavior of other entities in the neighborhood. Spatial relationships are fundamental for knowledge discovery in geographic databases and are strongly related to the discovered patterns. Knowledge discovery is a user-dependent process, but the user is usually neither an expert in geographic databases nor in spatial relationships. This paper presents an approach to reduce the number of spatial relationships for knowledge discovery, using a geo-ontology and semantic spatial integrity constraints. We show how spatial constraints can help the user of knowledge discovery in both defining the semantically consistent spatial relationships and reducing the verification of unnecessary relationships.

1 Introduction

The increasing use of geographic data in different application domains has resulted in large amounts of data stored in geographic databases. Geographic data are real world entities, also called spatial features [1], which have a location on Earth's surface. All spatial features (e.g. Portugal, Spain) belong to a feature type (e.g. country), and have both non-spatial attributes (e.g. name, population) and spatial attributes (geographic coordinates *x,y*). Figure 1 shows an example of spatial feature types, where shape is a spatial attribute characterizing the geometric representation (e.g. point, line or polygon), and the map is a graphical representation of some shapes.

Beyond the spatial attributes, there are implicit spatial relationships, which are intrinsic to geographic data, but usually not explicitly stored in geographic databases (e.g. Roads *cross* Rivers). Because of spatial relationships real world entities can affect the behavior of other features in the neighborhood. These implicit correlations can only be discovered with specific techniques for knowledge discovery.

Knowledge discovery in databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns from data [2]. In geographic databases knowledge discovery is the extraction of interesting spatial patterns and features, general relationships between spatial and non-spatial data, and other general characteristics of data not explicitly stored in these databases [3].

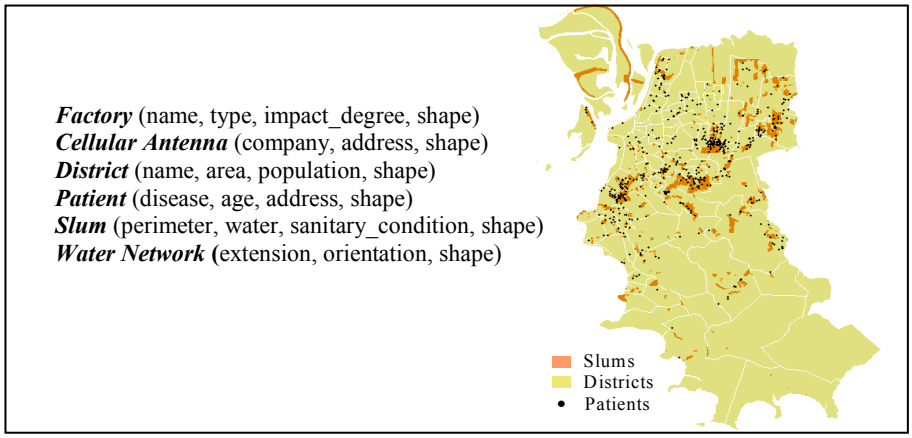


Fig. 1. Spatial feature types and its graphical representation (map)

Spatial join is the operation to compute spatial relationships between two spatial features. It is the most expensive operation in geographic databases for both spatial data analysis and knowledge discovery.

The algorithms for knowledge discovery are not intelligent enough to decide which aspects in geographic databases are relevant or not to the discovery process. The relationships and many other parameters should be provided by the KDD user, what makes the discovery process extremely user-dependent. However, the KDD user is usually not an expert in geographic databases, and he may not have enough background knowledge to decide which aspects to consider in the discovery process.

Geographic data share a large number of spatial relationships, but many are irrelevant to the discovery process and are unnecessarily calculated. For example, an *island* is a piece of land surrounded by *water*. In a geographic database, *island* should be represented as a spatial feature type with a mandatory relationship with a spatial feature type *water*. So, why should we compute spatial relationships between islands and water resources for knowledge discovery if by definition they are related to each other? Why should we consider this kind of relationships if they will create patterns with high confidence without adding novel knowledge? These and other aspects are usually not familiar to the KDD user, but are well-known concepts to geographers or geographic database designers.

Geographic database designers or specialists in Geography know the nature, the concepts, and the semantics of geographic data, so they are able to specify both mandatory and prohibited spatial relationships which define spatial integrity constraints. By specifying these constraints in a geo-ontology, the knowledge of specialists in geographic data can be reused to help the KDD user.

In the literature, there are basically two approaches for knowledge discovery in geographic databases: one is based on quantitative reasoning, which mainly computes distance relationships; and the other is based on qualitative reasoning. Algorithms based on qualitative reasoning [3,4,5,6,7] compute spatial relationships according to a relationships hierarchy, but they neither filter the relationships nor consider if they are geometrically possible or semantically consistent.

In this paper we show how to reduce the number of topological relationships for knowledge discovery in geographic databases with spatial integrity constraints and geo-ontologies. The novelty of our approach is the use of geo-ontologies as prior

knowledge to eliminate mandatory as well as prohibited topological relationships expressed by spatial integrity constraints, and deduce which topological relations may lead to interesting patterns in the KDD process.

The remainder of the paper is organized as follows: Section 2 presents the basic concepts of spatial relationships and spatial constraints. Section 3 presents a geo-ontology meta-model for geographic data and spatial integrity constraints. Section 4 shows how geo-ontologies and spatial integrity constraints can be used as prior knowledge to reduce geographic data pre-processing for knowledge discovery. Finally, Section 5 concludes the paper and suggests some directions of future work.

2 Spatial Relationships and Semantic Integrity Constraints

Geographic data share basically 3 types of spatial relationships: *direction, distance,* and *topological. Direction* relationships deal with the order as spatial features are located in space. *Distance* relations are based on the Euclidean distance between two spatial features. Our focus in this paper is on *topological* relations, which describe concepts of adjacency, containment and intersection between two spatial features.

There are many approaches in the literature to formally define a set of topological relationships among points, lines and polygons [8,9]. The OGC (Open GIS Consortium) [10], which is an organization dedicated for developing standards for spatial operations and spatial data interchange to provide interoperability between Geographic Information Systems (GIS), defines a standard set of topological operations: *disjoint, overlaps, touches, contains, within, crosses* and *equals*.

Considering the geometric representation of spatial features, different topological relationships are applicable. Table 1 shows the topological relationships, standardized by the OGC, considering the **geometry** of two spatial feature types. Empty boxes and checked boxes respectively represent impossible and possible relationships between two geometries. For example, two spatial features represented as line and polygon, respectively, can share the relationships *disjoint, touches, within* and *crosses.*

Table 1. Topological relationships between points, lines and polygons [10]

Topological Relation / Geometric Combination	Disjoint	Overlaps	Touches	Contains	Within	Crosses	Equals
Point(•) Point(•)	✓			✓	✓		✓
Point(•) Line(/)	✓		✓		✓	✓	
Point(•) Polygon(□)	✓		✓		✓	✓	
Line(/) Line(/)	✓	✓	✓	✓	✓	✓	✓
Line(/) Polygon(□)	✓		✓		✓	✓	
Polygon(□) Polygon(□)	✓	✓	✓	✓	✓		✓

Spatial integrity constraints encompass the peculiarities of geographic data and spatial relationships. Their purpose is to warrant as well as to maintain both the quality and the consistency of spatial features in geographic databases. Cockroft [11] proposed three types of spatial integrity constraints: topological, semantic, and user defined constraints. Topological integrity constraints refer to the topological consistency of the shape, such as "the boundary of a state must be contained inside

the shape of the country". Semantic constraints refer to the spatial consistency of spatial features according to their meaning (e.g. "lakes cannot contain rivers"). User defined integrity constraints are equivalent of "business rules" defined in non-geographic databases, such as, "residential areas must lie farther than 1000 meters from a nuclear plant".

Serviane [12] presented topological-semantic integrity constraints, which define mandatory or prohibited topological relationships according to the semantic of the spatial feature. Considering only the geometric representation of spatial features most topological relationships are possible. Considering their meaning, it is possible to define which topological relation is consistent and which one is inconsistent. Extending the approach to specify topological-semantic constraints proposed by Bogorny [13], in order to support the cardinality "all", for mandatory *disjoint* relationships, a topological-semantic constraint between two spatial feature types *A* and *B* can be defined as:

<constraint> ::= <spatialFeatureTypeA><predicate> <spatialFeatureTypeB>
<predicate> ::= <relType> <minCard> <maxCard>
<relType>::='touches'|'overlaps'|'equals'|'within'|'contains'|'crosses'|'disjoint'
<minCard> ::= 0|1| a
<maxCard> ::= 0|1| a

The predicate of a spatial constraint is given by a relationship type *relType,* a minimum cardinality *<minCard>*, and a maximum cardinality *<maxCard>*. The predicate can express mandatory constraints, which are given by the cardinalities (*a,a*) for the relationship *disjoint*, and (1,1) for the remaining topological relationships. A spatial constraint for Hospital with Factory, for example, can be defined as <Hospital> *<disjoint><a><a>*<Factory>, where all instances of Hospital are *disjoint* to ALL instances of Factory. A spatial constraint for Island with Water Resource, for example, where every Island has a *within* relationship with only one Water Resource can be expressed such as: <Island> *<within>* <1><1> <Water Resource>.

Prohibited constraints are defined through the cardinalities (0,0). For example, <River> *<contains>* <0><0> <Road>.

3 Geo-Ontologies

Ontology is an explicit specification of a conceptualization [14]. More specifically, ontology is a logic theory corresponding to the intentional meaning of a formal vocabulary, that is, an ontological commitment with a specific conceptualization of the world [15]. It is an agreement of both the concepts meaning and the structure of a specific domain. Each concept definition must be unique, clear, complete, and non-ambiguous. The structure represents the properties of the concept, including a description, attributes and relationships with others concepts.

Ontologies have been used recently in many and different fields in Computer Science, such as Artificial Intelligence, Databases, Conceptual Modeling, Semantic Web, etc. Although research is not so far yet in ontologies for geographic data [16], some geo-ontologies have been emerging recently. Besides defining a geo-ontology for administrative data for the country of Portugal, Chaves [17] defines a geo-ontology meta-model, named GKB (Geographic Knowledge Base).

GKB provides the concept of spatial *Feature,* which is represented as a class, and is associated to a *Feature_Type*, whose instances represent all feature types specified for a domain. For example, Country is an instance of *Feature_Type,* while Brazil and Portugal are instances of *Feature*. The class *Name* has names identified for every feature in all available information sources, including synonyms. Concepts of relationships among features in GKB are specified through the classes *Relationship* and *Relationship_Type,* which can assume concepts of *partOf* and *adjacency*.

In our point of view, a geo-ontology should provide the definition of the main aspects of geographic data, which are already defined in geographic meta-models for conceptual modeling (e.g. MADS, OMT-G) and standardized by the OGC. Based on these definitions, a geographic concept should have, at least: one spatial attribute given by a geometry, non-spatial attributes, relationships with other geographic concepts, and spatial constraints. The relationships can be conventional, such as aggregations or associations, or spatial, such as topological, distance or order. Considering these characteristics, we extended the GKB proposed in [17] to support geometry and spatial integrity constraints.

Figure 3 shows the extended GKB meta-model. The classes *GM_Object* and *GM_ObjectType* were added following the OGC definitions. *GM_ObjectType* represents the geometric representation of a feature type (e.g. point, line, and polygon). *GM_Object* is an instance of a geometric type associated to a specific feature. The cardinalities 0, 1, and *a* added to the dual relationship between the classes *Relationship* and *Feature* define concepts of mandatory or prohibited constraints.

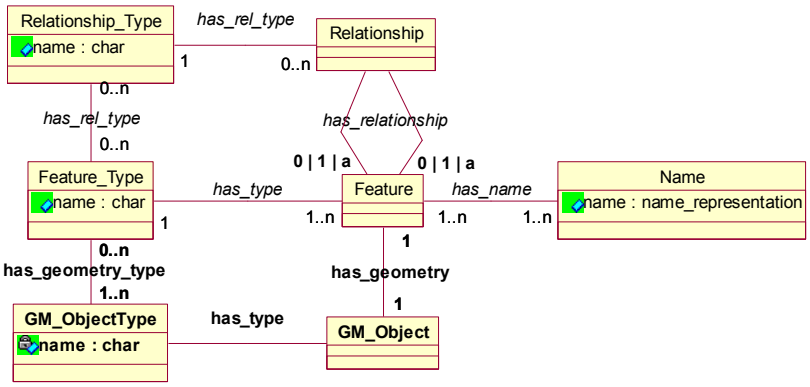


Fig. 3. Extended GKB to support geometry, topological relationships and spatial constraints

4 Geo-Ontologies and the KDD Process

The possible binary topological relationships between two geometric objects shown in Table 1 can be significantly reduced if we consider the semantics of each object. Table 2 shows an example of the same geometric combinations illustrated in Table 1, giving a different semantics to each geometric object. The geometries point and line, for example, can share the relationships *disjoint, touches, within* and *crosses* (see Table 1). Considering that point and line have respectively the semantics of Bridge

and River (see Table 2), then only *crosses* is semantically consistent. The combinations line/line, for example, can share any topological relation, but if their semantics is respectively River and Road, then only *disjoint, touches* and *crosses* are consistent (see Table 2). For the combination polygon/polygon with the semantics State and Country respectively, only the relationship *within* is consistent.

Table 2. Possible topological relationships considering the semantics of the feature types

Topological Relation / Semantic Combinations	Disjoint	Overlaps	Touches	Contains	Within	Crosses	Equals
Factory (•) Hospital (•)	✓						
Bridge (•) River (/)						✓	
Factory (•) Airport(□)	✓		✓				
River (/) Road (/)	✓		✓			✓	
Beach (/) Sea (□)			✓				
State (□) Country (□)					✓		

Although the topological relationships shown in Table 2 are semantically possible, not all of them are interesting for knowledge discovery. So, if beside considering the semantics of the features we also consider spatial integrity constraints, it is possible to reduce still more the number of spatial joins and define which relationships should be computed for knowledge discovery.

Applying spatial integrity constraints, Table 3 shows the possible topological relationships between the same feature types shown in Table 2. Considering only the semantics of the spatial feature types, we would have 9 possible relationships according to the example shown in Table 2. Considering spatial integrity constraints we would have only 3 relevant relationships to consider in the discovery process.

Table 3. Topological relationships for knowledge discovery

Topological Relation / Semantic Combinations	Disjoint	Overlaps	Touches	Contains	Within	Crosses	Equals
Factory (•) Hospital (•)							
Bridge (•) River (/)							
Factory (•) Airport(□)			✓				
River (/) Road (/)			✓			✓	
Beach (/) Sea (□)							
State (□) Country (□)							

On the one hand, the prohibited constraints forbid the inconsistent relationships, so they should not exist in the database. By consequence, they do not need to be computed for spatial analysis or knowledge discovery. On the other hand, mandatory relationships will produce patterns with high confidence in the discovery process because mandatory relationships will always hold if the database is consistent. However, these patterns will not add novel knowledge to the discovery.

Despite mandatory and prohibited constraints do not explicitly define the relevant relationships for knowledge discovery, we are able to eliminate those which are

mandatory or prohibited, and specify those which are possible. Let us consider the set of all topological relationships as R = {*touches, contains, within, crosses, overlaps, equals, disjoint*}. T is the set of topological relationships **geometrically** possible between two feature types A and B. Pr is the set of **prohibited** relationships between A and B, M is the set of **mandatory** relationships and P_{KDD} is the set of **possible** relationships for knowledge discovery. If a prohibited constraint is given between A and B, then the set of possible relationships is $P_{KDD(A,B)} = T_{(A,B)} - Pr_{(A,B)}$. If a mandatory constraint is defined between A and B, then $P_{KDD(A,B)} = \phi$.

The approximate reduction cost of computing spatial joins for each pair of spatial feature types A and B for knowledge discovery is given by $R_{cost(A,B)} = (|T_{(A,B)}| - |P_{KDD(A,B)}|)$. $Cost_{re(A,B)}$, where $Cost_{re\ (A,B)}$ is the time to compute each topological relationship between A and B. The cost to browse the geo-ontology is not considered.

In the discovery process, a data pre-processing algorithm can compute the topological relationships according to the properties of the feature types specified in the geo-ontology. For example, let us consider that the feature type of interest specified by the KDD user is *River* and that the relevant feature types to be spatially compared with *River* are *Road*, *Hospital*, and *Island*. Suppose that in a geo-ontology *River* has the properties of a mandatory relationship *disjoint* with *Hospital* and a prohibited relationship *contains*, *overlaps*, *inside* and *equals* with *Road*, but no property with *Island*. The first step of the pre-processing algorithm is to read the properties of *River* and specify that $P_{KDD(River,Road)}$ = {*touches, crosses*} and $P_{KDD(River,Hospital)} = \phi$. As P_{KDD} is already defined for *Road* and *Hospital*, the second step is to read the properties of *Island* in the geo-ontology, and specify $P_{KDD(River,Island)}$. Suppose that *Island* has the property of a mandatory relationship *within*, with *River*, than $P_{KDD(River,Island)} = \phi$.

5 Conclusions and Future Work

In this paper we presented a geo-ontology meta-model to define concepts and properties of geographic data. Through the properties we can specify spatial integrity constraints, which forbid or obligate specific topological relationships between specific feature types.

Considering only the geometry of spatial feature types, a certain number of topological relationships is possible. We showed how this number can be reduced if we consider the semantics of the spatial features and their spatial integrity constraints, using geo-ontologies. We also showed how the spatial integrity constraints can contribute for knowledge discovery in geographic databases. The mandatory and the prohibited spatial relationships defined by the constraints are irrelevant to the discovery process because of two reasons: - *prohibited relationships* will never exist if the database is consistent; and - *mandatory relationships* will produce patterns with high confidence but which do not add any novel knowledge to the discovery process.

As future work, we will study the application of distance and order constraints and how we can reduce the number of spatial joins for the KDD process with different combinations of spatial relationships.

6 ACKNOWLEDGMENTS

We would like to thank CAPES and CNPQ for the financial support of this research, and Stefano Spaccapietra and Daniela Leal Musa, for their comments.

7 References

1. Opengis. The Opengis Abstract Specification Topic 1: Feature Geometry. In URL: Http://Www.Opengeospatial.Org/Docs/01-101.Pdf (2001).
2. Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P.: Knowledge Discovery and Data Mining: Towards a Unifying Framework. In: International Conference on Knowledge Discovery and Data Mining (KDD'96). AAAI Press, Portland, OR, (1996) 82-88.
3. Lu, W., Han, J., Ooi, B. C.: Discovery of General Knowledge In Large Spatial Databases. Workshop on Geographic Information Systems, Singapore (1993) 275-289.
4. Ester, M., Frommelt, A., Kriegel, H.-P., Sander, J.: Spatial Data Mining: Database Primitives, Algorithms And Efficient DBMS Support. Journal of Data Mining and Knowledge Discovery. 4 (2000) 193-216.
5. Malerba, D., Appice, A., Vacca N.: SDMOQL: An OQL-Based Data Mining Query Language For Map Interpretation Tasks. In: Workshop On Database Technologies For Data Mining. Springer, Prague, Czech Republic (2002).
6. Appice, A., Ceci, M., Lanza, A.: Discovery of Spatial Association Rules In Geo-Referenced Census Data: A Relational Mining Approach. Intelligent Data Analysis. Software & Data 6 (2003).
7. Koperski, K., Han, J.: Discovery of Spatial Association Rules In Geographic Information Databases. In: International Symposium In Large Spatial Databases. Springer, Portland, Maine, USA (1995) 47-66.
8. Clementini, E., Di Felice, P.: A Model for Representing Topological Relationships between Complex Geometric Features In Geographical Databases. Information Sciences. 90 (1996) 121-136.
9. Egenhofer, M., Herring, J.: Categorizing Binary Topological Relations Between Regions, Lines, and Points In Geographic Databases. Technical Report TR-941, University of Maine, (1994).
10. Opengis. Open GIS Simple Features Specification For SQL. In URL: Http://Www.Opengeospatial.Org/Docs/99-054.Pdf (1999).
11. Cockcroft, S.: A Taxonomy of Spatial Data Integrity Constraints. Geoinformatica, Kluwer Academic Publishers, Hingham, MA, USA. 1-4 (1997) 327-343.
12. Servigne, S. et al.: A Methodology for Spatial Consistency Improvement of Geographic Databases. Geoinformatica. 4-1 (2000) 7-34.
13. Bogorny, V., Iochpe, C.: Extending the OpenGIS Model to Support Topological Integrity Constraints. In: Brazilian Symposium on Databases, COPPE/UFRJ, Rio de Janeiro, Brazil (2001) 25-39 (in Portuguese).
14. Gruber, T. R.: Towards Principles for the Design of Ontologies Used for Knowledge Sharing. Formal Ontology in Conceptual Analysis and Knowledge Representation. Int. J. of Human-Computer Studies. Kluwer Academic Publishers. 43 (1993) 907-928.
15. Guarino, N.: Formal Ontology and Information Systems. In: International Conference on Formal Ontology in Information Systems. Italy (1998) 3-15.
16. Spaccapietra, S., Cullot, N., Parent, C., Vangenot, C.: On Spatial Ontologies. In: Brazilian Symposium on GeoInformatics. Campos do Jordão, Brazil. (2004).
17. Chaves, M. S., Silva, M. J., Martins, B.: A Geographic Knowledge Base for SemanticWeb Applications. In Brazilian Symposium on Databases, Uberlandia, Minas Gerais, Brazil (2005). To appear.

Using Taxonomies to Facilitate the Analysis of the Association Rules

Marcos Aurélio Domingues[1] and Solange Oliveira Rezende[2]

[1] LIACC-NIAAD – Universidade do Porto
Rua de Ceuta, 118, Andar 6 – 4050-190 Porto, Portugal
marcos@liacc.up.pt
[2] Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
Av. Trabalhador São-Carlense, 400, Cx. Postal 668 – 13560-970 São Carlos, SP, Brazil
solange@icmc.usp.br

Abstract. The Data Mining process enables the end users to analyse, understand and use the extracted knowledge in an intelligent system or to support in the decision-making processes. However, many algorithms used in the process encounter large quantities of patterns, complicating the analysis of the patterns. This fact occurs with association rules, a Data Mining technique that tries to identify intrinsic patterns in large data sets. A method that can help the analysis of the association rules is the use of taxonomies in the step of post-processing knowledge. In this paper, the $\mathcal{GART}$ algorithm is proposed, which uses taxonomies to generalize association rules, and the *RulEE-GAR* computational module, that enables the analysis of the generalized rules.

1 Introduction

The development of the data storing technologies has increased the data storage capacity of companies. Nowadays the companies have technology to store detailed information about each performed transaction, generating large databases. This stored information may help the companies to improve themselves and because of this the companies have sponsored researches and the development of tools to analyse the databases and generate useful information.

During years, manual methods had been used to convert data in knowledge. However, the use of these methods has become expensive, time consuming, subjective and non-viable when applied at large databases.

The problems with the manual methods stimulated the development of processes of automatic analysis, like the process of Knowledge Discovery in Databases or Data Mining. This process is defined as a process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data [6].

In the Data Mining process, the use of the association rules technique may generate large quantities of patterns. This technique has caught the attention of companies and research centers [3]. Several researches have been developed with this technique and the results are used by the companies to improve their

businesses (insurance policy, health policy, geo-processing, molecular biology) [8, 4, 9].

A way to solve the problem of the large quantities of patterns extracted by the association rules technique is the use of taxonomies in the step of post-processing knowledge [1, 8, 10]. The taxonomies may be used to prune uninteresting and/or redundant rules (patterns) [1].

In this paper the $\mathcal{GART}$ algorithm and the *RulEE-GAR* computational module is proposed. The $\mathcal{GART}$ algorithm (*Generalization of Association Rules using Taxonomies*) uses taxonomies to generalize association rules. The *RulEE-GAR* computational module uses the $\mathcal{GART}$ algorithm, to generalize association rules, and provides several means to analyze the generalized rules.

This paper is organized as following: first by presenting the association rules technique and some general features about the use of taxonomies, second by describing the $\mathcal{GART}$ algorithm and the *RulEE-GAR* computational module. Finally the results of some experiments performed with the $\mathcal{GART}$ algorithm along with our conclusion are presented.

2 Association Rules and Taxonomies

An association rule $LHS \Rightarrow RHS$ represents a relationship between the sets of items LHS and RHS [2]. Each item I is an atom representing the presence of a particular object. The relation is characterized by two measures: support and confidence. The support of a rule R within a dataset D, where D itself is a collection of sets of items (or itemsets), is the number of transaction in D that contain all the elements in $LHS \cup RHS$. The confidence of the rule is the proportion of transactions that contain $LHS \cup RHS$ with respect to the number of transactions that contain LHS. The problem of mining association rules is to generate all association rules that have support and confidence greater than the minimum support and minimum confidence defined by the user to mine association rules. High values of minimum support and minimum confidence just generate trivial rules. Low values of minimum support and minimum confidence generate large quantities of rules (patterns), complicating the user's analysis.

A way of overcoming the difficulties in the analysis of large quantities of association rules is the use of taxonomies in the step of post-processing knowledge. The use of taxonomies may help the user to identify interesting and useful knowledge in the extracted rules set. The taxonomies represent a collective or individual characterization of how the items can be classified hierarchically [1]. In Fig. 1 an example of a taxonomy is presented where it can be observed that: *t-shirts* are *light clothes*, *shorts* are *light clothes*, *light clothes* are a kind of *sport clothes*, *sandals* are a kind of *shoes*.

In the literature there are several algorithms to generate association rules using taxonomies (generalized association rules). Algorithms like *Cumulate* and *Stratify* [10] generate rules sets larger than rules sets generated without taxonomies (because they generate association rules with and without taxonomies). To try decrease the quantity of generated rules, a subjective measure is used to

prune the uninteresting rules [10]. The subjective measure does not guarantee that the quantity of rules will decrease. Our method proposes an algorithm and a module of post-processing [5]. Using the module, the user looks to a small set of rules without taxonomies, builds some taxonomies and then uses the algorithm to generalize the association rules, pruning the original rules that are generalized. Thus our algorithm always decreases or keeps the volume of the rules sets. The proposed algorithm and module are presented in Section 3 and 4.

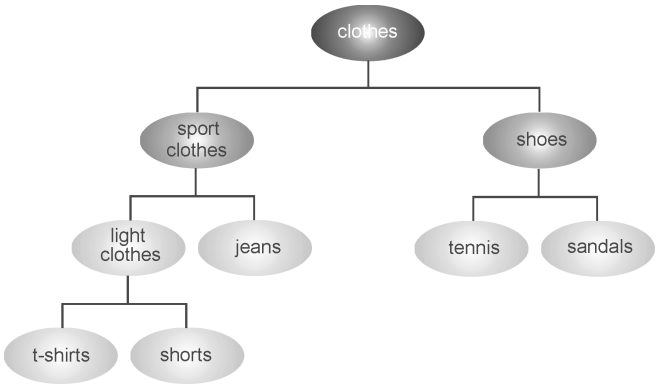


Fig. 1. An example of taxonomy for clothes.

3 The Algorithm $\mathcal{GART}$

We analysed the structure of the association rules generated by algorithms that do not use taxonomies. The results of the analysis show us that it is possible to generalize association rules using taxonomies. In Fig. 2 we show how the association rules can be generalized.

First we changed the items *t-shirt* and *short* of the rules *short & slipper* $\Rightarrow$ *cap*, *sandal & short* $\Rightarrow$ *cap*, *sandal & t-shirt* $\Rightarrow$ *cap* and *slipper & t-shirt* $\Rightarrow$ *cap* by the item *light clothes* (which represents a generalization). This change generated two rules *light clothes & slipper* $\Rightarrow$ *cap* and two rules *light clothes & sandal* $\Rightarrow$ *cap*. Next, we pruned the repeated generalized rules, maintaining only the two rules: *light clothes & slipper* $\Rightarrow$ *cap* and *light clothes & sandal* $\Rightarrow$ *cap*.

The two rules generated by the Step 1 (Fig. 2) were generalized again. We changed the items *slipper* and *sandal* by the item *light shoes* (which represented another generalization) generating two rules *light clothes & light shoes* $\Rightarrow$ *cap*. Then we pruned the repeated generalized rules again, maintaining only one generalized association rule: *light clothes & light shoes* $\Rightarrow$ *cap*.

Due to the possibility of generalization of the association rules (Fig. 2), we propose an algorithm to generalize association rules. The proposed algorithm is illustrated in Fig. 3. We called the proposed algorithm of $\mathcal{GART}$ (*Generalization of Association Rules using Taxonomies*).

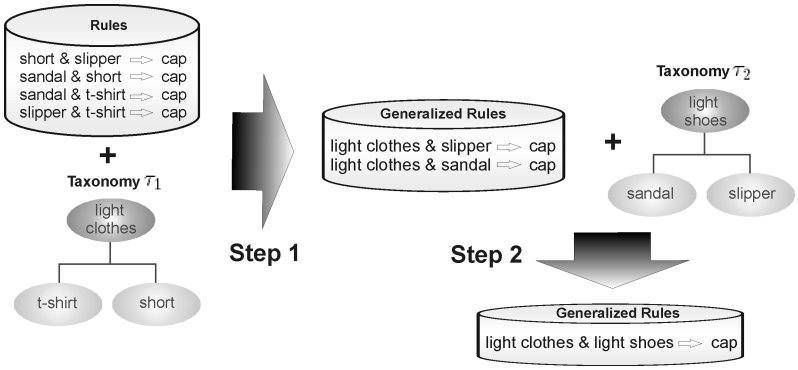


Fig. 2. Generalization of association rules using two taxonomies.

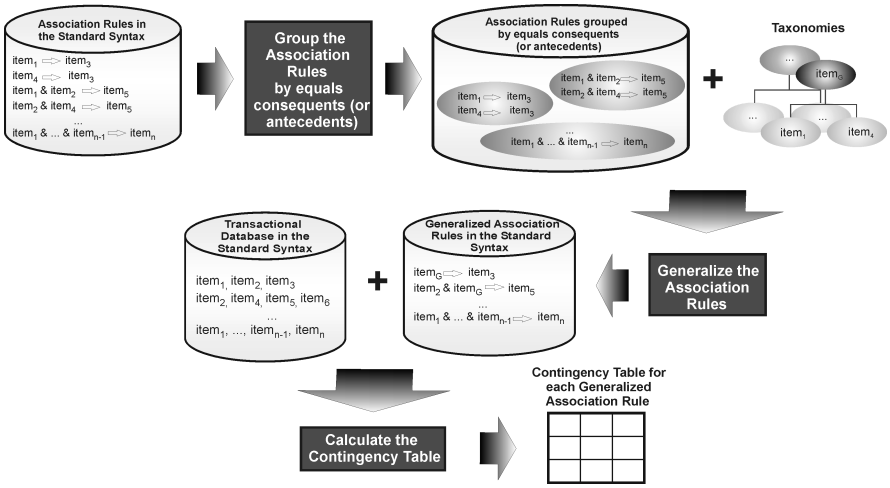


Fig. 3. The proposed algorithm to generalize association rules.

The proposed algorithm just generalizes one side of the association rules - LHS or RHS (after to look to a small set of rules without taxonomies, the user decides which side will be generalized). First, we grouped the rules in subsets that present equal antecedents or consequents. If the algorithm were used to generalize the left hand side of the rules (LHS), the subsets would be generated using the equals consequents (RHS). If the algorithm were used to generalize the right hand side of the rules (RHS), the subsets would be generated using the equal antecedents (LHS). Next, we used the taxonomies to generalize each subset (as illustrated in Fig. 2). In the final algorithm we stored the rules in a set of generalized association rules.

In the final algorithm, we also calculated the Contingency Table for each generalized association rules to get more information about the rules. The Contingency Table of a rule represents the coverage of the rule with respect to the database used in its mining [7]. With the calculation of the Contingency Table we finished the algorithm.

4 The Computational Module *RulEE-GAR*

In this section we present the *RulEE-GAR* computational module that provides means to generalize association rules and also to analyze the generalized rules [5]. The generalization of the association rules is performed by the $\mathcal{GART}$ algorithm, described in the previous section. Next we describe the means to analyze the generalized association rules. In Fig. 4 we showed the screen of the interface that enables the user to analyze and to explore the generalized rules sets.

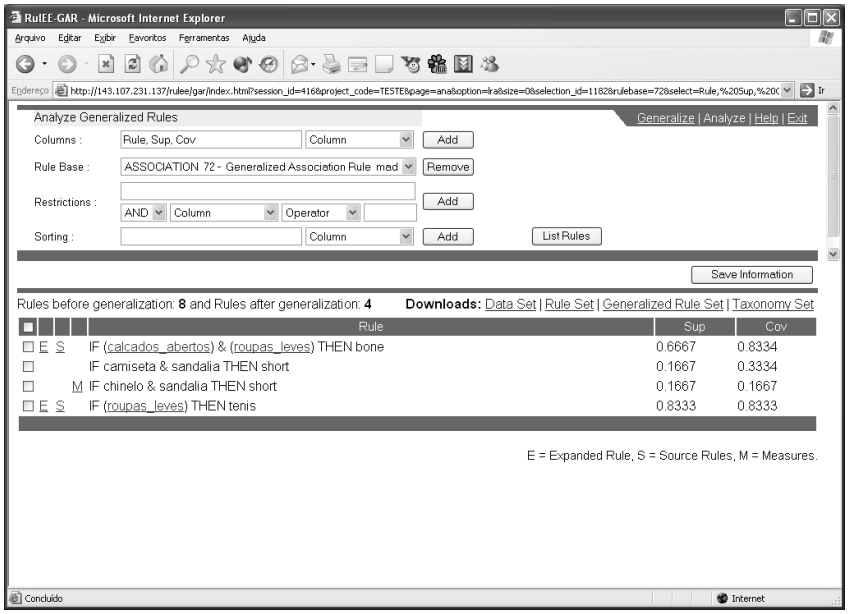


Fig. 4. Screen of the analysis interface of generalized association rules.

On the screen of the analysis interface of generalized rules (Fig. 4) there are some spaces where the user puts data to make a query and select a set of generalized rules, accompanied or not of several evaluation measures [7], to be analyzed. Besides allowing the user to select a set of rules, the interface provides four links in the section *Downloads* to look for and/or download the files. The files contain, respectively, the set of transactional data (*Data Set*), the set of

source rules (*Rule Set*), the set of generalized rules (*Generalized Rule Set*) and the set of taxonomies used to generalize the rules (*Taxonomy Set*).

Besides links for visualization and/or download of the files, each generalized association rule presents others links that enable the user to explorer information about the generalization of the rule. The links are positioned at the left side of the rules (Fig. 4). The links are described as following:

Expanded Rule It is represented in the interface by the letter "E". This link enables the user to see the generalized rule in expanded way. The generalized items of a rule are changed by the respective specific items.

Source Rules It is represented in the interface by the letter "S". This link enables the user to see the source rules that were generalized.

Measures It is represented in the interface by the letter "M". This link is available only if the user selects the support (*Sup*) and/or confidence (*Cov*) measures in its query and these measures present values lower than the minimum support and/or minimum confidence values defined to the mining process of the rules set not generalized. With this link it is possible to see which generalized rules have support and/or confidence values lower than the minimum support and/or minimum confidence values.

In Fig. 4 we also see that the generalized items in a rule (items between parentheses) are presented as links. These links enable the user to see the source items that were generalized. In the analysis interface, the user can also store the information, selected by the query, in a text file.

5 Experiments

We performed some experiments using the $\mathcal{GART}$ algorithm to demonstrate that the use of taxonomies, to generalize large rules sets, reduces large quantities of association rules and makes easy the analysis of the rules.

The experiments were performed using a sale database of a Brazilian supermarket. The database contained sales data of the recent 3 month. We made 4 partitions of the database to perform the experiments. The partitions were made using the sale data along of 1 day, 7 days, 14 days and 1 month.

To generate the association rules, we used the implementation of the *Apriori* algorithm performed by Chistian Borgelt[3] with minimum support value equal 0.5, minimum confidence value equal 0.5 and a maximum number of 5 items by rule. The generated rules sets are described as following:

- RuleSet_1day - 32668 rules generated using the partition of 1 day;
- RuleSet_7days - 19166 rules generated using the partition of 7 days;
- RuleSet_14days - 16053 rules generated using the partition of 14 days;
- RuleSet_1month - 21505 rules generated using the partition of 1 month;

[3] Available for downloading at the web site `http://fuzzy.cs.uni-magdeburg.de/~borgelt/software.html`.

– RuleSet_3months - 19936 rules generated using the whole database (3 months of sale data).

To perform the experiments, we looked to the database and to the 5 sets of association rules generated to make 18 sets of taxonomies. Then we ran the $\mathcal{GART}$ algorithm combining each set of taxonomies with each set of rules. In Fig. 5 a chart is presented that shows the reduction rates of the 5 rules sets after running $\mathcal{GART}$ algorithm using the 18 sets of taxonomies to generalize each rules set. In Fig. 5 the sets of taxonomies are called "T" followed by an identification number, as for example: T01.

As it can be observed in Fig. 5, the experiments show reduction rates of the sets of association rules varying from 14,61% to 50,11%.

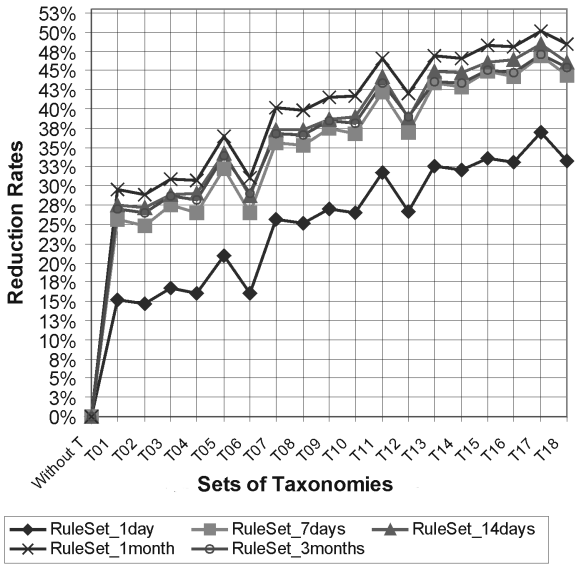


Fig. 5. Reduction rates got using taxonomies to generalize association rules.

6 Conclusion

A problem found in the Data Mining process is the fact that several of the used algorithms generate large quantities of patterns, complicating the analysis of the patterns. This problem occurs with the association rules, a Data Mining technique that tries to identify all the patterns in a database.

The use of taxonomies, in the step of knowledge post-processing, to generalize and to prune uninteresting and/or redundant rules may help the user to analyze the generated association rules.

In this paper we proposed the $\mathcal{GART}$ algorithm that uses taxonomies to generalize association rules. We also proposed the *RulEE-GAR* computational module that uses the $\mathcal{GART}$ algorithm to generalize association rules and provides several means to analyse the generalized association rules. Then we presented the results of some experiments performed to demonstrate that the $\mathcal{GART}$ algorithm may reduce the volume of the sets of association rules. As the sets of taxonomies were made by the user, others sets of taxonomies may generate reduction rates higher than the rates presented in our experiments, mainly whether the sets were made by experts in the application domain.

Acknowledgements. This work was supported by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), Brazil.

References

1. J. M. Adamo. *Data Mining for Association Rules and Sequential Patterns*. Springer-Verlag, New York, NY, 2001.
2. R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo, editors, *Proceedings of Twentieth International Conference on Very Large Data Bases, VLDB*, pages 487–499, 1994.
3. B. Baesens, S. Viaene, and J. Vanthienen. Post-processing of association rules. In *Proceedings of the Special Workshop on Post-Processing. The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2–8, 2000.
4. E. Clementini, P. Di Felice, and K. Koperski. Mining multiple-level spatial association rules for objects with a broad boundary. *Data & Knowledge Engineering*, 34(3):251–270, 2000.
5. M. A. Domingues. Generalização de regras de associação, 2004. Masters Thesis, Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, São Carlos, SP - Brazil.
6. U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34, 1996.
7. N. Lavrac, P. Flach, and R. Zupan. Rule evaluation measures: A unifying view. In S. Dzeroski and P. Flach, editors, *Proceedings of the Ninth International Workshop on Inductive Logic Programming (ILP-99)*, volume 1634, pages 174–185. Springer-Verlag, 1999. LNAI.
8. B. Liu, W. Hsu, S. Chen, and Y. Ma. Analyzing the subjective interestingness of association rules. *IEEE Intelligent Systems & their Applications*, 15(5):47–55, 2000.
9. T. Semenova, M. Hegland, W. Graco, and G. Williams. Effectiveness of mining association rules for identifying trends in large health databases. In *Workshop on Integrating Data Mining and Knowledge Management. ICDM'01: The 2001 IEEE International Conference on Data Mining*, 2001. Avaliable in `http://cui.unige.ch/~hilario/icdm-01/DM-KM-Final/Semenova.pdf`. Access in 11/01/2005.
10. R. Srikant and R. Agrawal. Mining generalized association rules. *Future Generation Computer Systems*, 13(2–3):161–180, 1997.

Improving Classification of Multi-Lingual Web Documents using Domain Ontologies

Marina Litvak, Mark Last, and Slava Kisilevich

Department of Information Systems Engineering, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel,
{litvakm, mlast, slaks}@bgu.ac.il

Abstract. In this paper, we deal with the problem of analyzing and classifying web documents to several major categories/classes in a given domain using domain ontology. We present the ontology-based web content mining methodology that contains such main stages as collecting a training set of labeled documents from a given domain, building a classification model above this domain given the domain ontology, and classification of new documents via the induced model. We tested the proposed methodology in a specific domain, namely web pages containing information about production of certain chemicals. Using our methodology, we are interested to identify all relevant web documents while ignoring the documents that do not contain any relevant information. Our system receives as input an OWL file built in Protege tool, which contains the domain-specific ontology, and a set of web documents classified by a human expert as "relevant" or "non-relevant". We use a language-independent key-phrase extractor with integrated ontology parser (defined in a given language) for creating the database from input documents and use it as a training set for the classification algorithm. The system classification accuracy using various levels of ontology is evaluated.The current version of our system supports web content mining in English, Arabic, Russian, and Hebrew languages.

1 Introduction

Over the last years, we have observed an explosive growth in the information available on the Web. To meet our information needs, we need more intelligent systems to gather the useful information from the huge amount of Web related data sources.

Web mining ([2]) is a new technology that has emerged as a popular area in the field of Web Intelligence ([4]). It is categorized into three areas of interest: web usage mining (finds access patterns from web logs), web structure mining (provides structural information about documents) and web content mining (finds useful information from the web content) [1]. It is obvious that data mining techniques (see [5], [6]) can be used for Web mining. One of the problems in this area is to represent the web documents as a meaningful, informative input for data mining algorithms, and then to "translate"/interpret the mining results.

In this paper, we introduce the ontology-based web content mining application for analyzing and classifying web documents in a given domain. We use *domain ontology*, which organizes concepts, relations and instances into a domain [11], for purpose of enriching the term vectors representing documents with concepts. This approach has two benefits: first, it resolves synonyms; and second, it introduces more general concepts. Our term vectors contain of terms and their importance weights, where term may be a phrase extracted from the

text of a document or related concept from the ontology (depending on the level of concept hierarchy or abstraction level induced by the user). For the purpose of classification, we can use any popular classification algorithm, like C4.5, Bayes Network and Naive Bayes.

The rest of the paper is organized in the following way. Section 2 summarizes the related work. Section 3 describes the methodology and the proposed system. Section 4 depicts the tested domain and the constructed ontology. In Section 5, we evaluate the results of initial experiments. Finally, in the last section we outline the conclusions and the future work.

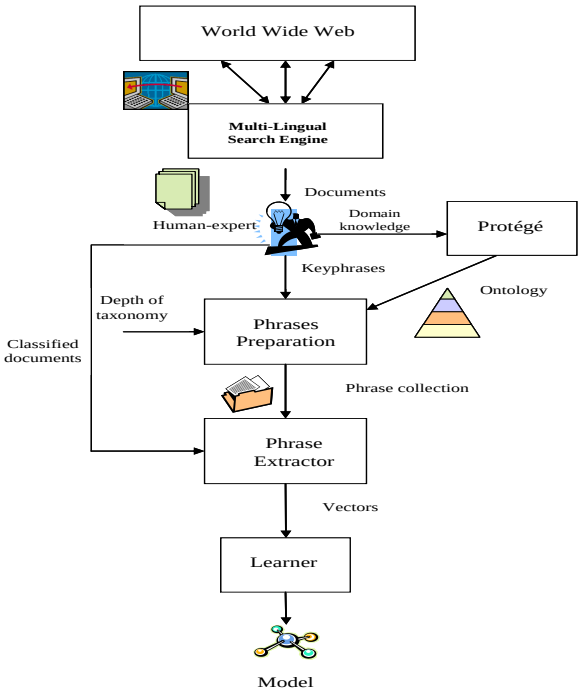


Fig. 1. Cross-Lingual Web Classification System

2 Related work

During the last decade, a huge amount of issues related to web content mining was investigated, like discovering of different patterns in the static content using conventional data mining [3], dynamic content mining (like mining news from online news sites) [7], predicting web information content [8], developing recommendation systems that can suggest the "information content" (IC) pages [9],

classifying web documents into Web hierarchy or topic ontology [20],[21], and many other.

Many authors reduce building recommendation systems to the classification task. Billsus and Pazzani [12] trained a Naive Bayes classifier [13] to recommend news stories to a user, using a Boolean feature vector representation of the candidate articles, where each feature indicates the presence or absence of a word in the article. Jennings and Higuchi [14] trained one neural network for each user to represent a user's preferences for news articles. Anderson and Horvitz [15] built a Naive Bayes model to predict the candidates (pages or topics) that the user will view next in the session.

Document representations for test classification are typically based on the classical Bag-Of-Words paradigm. However, over last years, the authors tried to enhance the classical document representation through concept-based document retrieval ([26]). One of such enhanced approaches is ontology[1].

Currently, there are several existing approaches for classifying web pages into Web hierarchy. Koller and Sahami in [20] propose an approach that utilizes the hierarchical topic structure to decompose the classification task into a set of simpler problems, one at each node in the classification tree. Mladenic and Globelnik in [21] describe an approach to automatically mapping web pages onto ontology using the Yahoo! ontology of Web pages. The paper of McCallum *et. al* ([22]) shows that the accuracy of a Naive Bayes text classifier can be significantly improved by taking advantage of a hierarchy of topic categories of documents. Chakrabarti *et. al* in [23] explore how to organize large text databases hierarchically by topic to aid better searching, browsing and filtering.

Bloehdorn and Hotho in [25] propose document representation through concepts extracted from background knowledge. In another publication ([24]) Hotho *et. al* use ontologies to improve text document clustering. A paper by Cesarano *et. al* [16] presents a prototype of an ontology-based system for information retrieval on the web, where the global relevance grade is computed for each document.

3 Methodology

Figure 1 presents a high-level view of the proposed Cross-Lingual Web Classification System. In the absence of any detailed domain knowledge, a user can initiate the system operation by submitting a set of keyword queries in any language to a multi-lingual search engine (such as Google^{TM}). A human expert reads the documents and labels them as "relevant" or "irrelevant". Additional degrees of relevancy (e.g., "partially relevant") can be allowed. The task of the Learner module is to build a compact model (profile) of the pages collected

[1] According to the most cited definition in the literature [10], ontology is an explicit specification of a domain conceptualization. It accumulates and organizes knowledge about domain in a machine-processable and human-readable way providing a common understanding basis, facilitating information/knowledge dissemination and reuse. Therefore, ontology has the potential to improve information/knowledge capturing, organization, re-use and re-finding through meticulous domain organization principles and advanced reasoning tasks.

from the web so that new relevant pages can be reliably recognized by the system. We induce a classification model from a training collection that includes a mix of relevant and non-relevant pages. Each page is represented as a vector of $< term_i,\ weight_i >$ pairs received from Ontology-based Phrase Extractor module, described in the sub-section below. The phrases are extracted using a list of domain-specific terms and other ontology information. The term-frequency (tf) $weight_i$ indicates the frequency of a $term_i$ in the observed document.

3.1 Ontology Specification

An ontology defines explicitly the terms used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information. Ontologies include computerusable definitions of basic concepts in the domain and the relationships among them. They encode knowledge in a domain and also knowledge that spans domains [19]. The term 'ontology' can be used for several ways. Ontologies can contain simple taxonomies and logical theories as well.

In this paper, an ontology represents the conceptual information of the domain of interest (see Section 4) and it is used for the purpose of conceptual document representation and improving the documents classification. In other words, our goal is extraction of more meaningful and relevant (even general) information from text of documents for the purpose of building more accurate classification models. Our ontology consists of individuals/instances, classes with their properties and hierarchical/taxonomic relationships between them. Each object/thing in the domain is associated with its unique class. Usually the names of classes are nouns. Each thing has a name (the name itself is not object of the domain but only symbolizes it) or several names that are synonyms. All names of an object are mapped to ontology as individuals of its class. The relationships among the things represent the existing taxonomy. The properties describe the things.

3.2 Ontology-based Phrase Extractor

This module includes Phrase Preparation and Phrase Extractor units (see Fig. 1). The module receives as input documents, ontology and abstraction level and creates term vectors.

The Phrase Preparation unit prepares phrase collection given ontology and abstraction level k — XML file including all general thing names as phrases with their associated classes of k^{th} level as related concepts (in case of abstraction level equal to 0 the collection does not include related concepts). Currently, we also add to this collection phrases that, by expert opinion, can characterize type of a document. In the future we are going to build a separate ontology containing these phrases or even embed them into an existing domain ontology.

The Phrase Extractor scans the phrases included in the collection, and every time it finds name of thing it references to the related concept. We used **Replace Terms by Concepts ("repl")** strategy (HYPINT) for replacing terms by

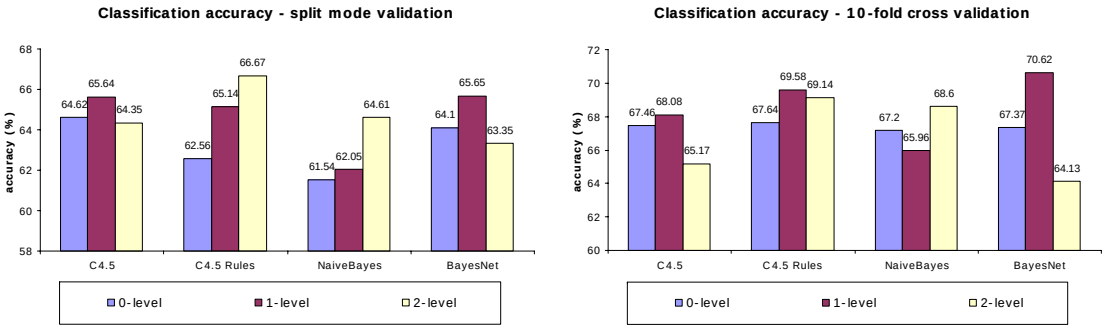
concepts and **All Concepts ("all")** strategy for disambiguation investigated in [24]. **Replace Terms by Concepts (repl)** strategy expels all terms from the vector representations for which at least one corresponding concept exists. Thus, terms that symbolize general things in domain ontology are only considered at the concept level, but terms that do not appear in ontology (provided directly by a human expert) are not discarded. The **All Concepts (all)** does not do anything about disambiguation and considers all concepts for augmenting the text document representation. The concept frequency is calculated as sum of the frequencies of all terms in document being related to that concept in the ontology.

The generic structure of this module enables to handle texts in virtually any language.

4 Experiments

The main goal of this research is increasing the classification accuracy through maintaining an ontology. We tested the proposed methodology in a specific domain, namely web pages containing information about production of certain chemicals. It is clear, that almost every chemical has many names (synonyms) - it may be a full name, an abbreviation, a formula or molecular structure. Our ontology stores class for each chemical in domain that contains all its known names as instances. Whenever the Phrase Extractor finds any name of chemical, it refers to the associated class. In addition, we define different properties for the chemicals and keep the hierarchical relationships between groups of them, like "available chemicals" (can be purchased or extracted from something), "rare chemicals" (complement to the first one), organic chemicals, salt, poisons and more. These groups may be joint as well as disjoint. The total time spent for ontology creation was about 20 hours including 2-3 meetings with a domain expert. Currently, our ontology includes 29 instances (things/names of chemicals) organized into 37 classes. We wish to extend it in the future experiments.

We learned and tested four classification models on the following document representations: vectors of original phrases (without any knowledge about concepts and relationships between them kept in the domain ontology), the same documents after phrase extraction with synonyms handling (1-level conceptualization), and after 2-level conceptualization (referring extractor every time it finds name of chemical to the parent classes of its thing class), and then compared between the accuracy rates of the resulting models. We were given 114 HTML pages classified as relevant and non-relevant by a domain expert (41 pages or 36% are relevant). Charts in Fig.2 demonstrate the classification accuracy of different models depending on level of ontology conceptualization. We applied C4.5, C4.5 Rules, Bayes Network and Naive Bayes algorithms using Weka Data Mining Software [18] and two testing modes: 10-fold cross-validation and test split (66% training set and remainder the testing set). The values in Fig.2 are averages from 10 runs of each mode. We used t-test (two-tailed paired, $\alpha = 0.05$) for each algorithm to assess whether the accuracy values of different abstraction



(a) (b)

Fig. 2. Classification Accuracy depending on Abstraction Level

levels	C4.5	C4.5 Rules	Naive Bayes	Bayes Network
0 – 1	↑ 0.037*	↑ 0.032*	↑ 0.738	↑ 0.336
1 – 2	↓ 0.340	↑ 0.455	↑ 0.053	↓ 0.124
0 – 2	↓ 0.832	↑ 0.104	↑ 0.164	↓ 0.480

levels	C4.5	C4.5 Rules	Naive Bayes	Bayes Network
0 – 1	↑ 0.411	↑ 0.039*	↓ 0.044*	↑ 0.001*
1 – 2	↓ 0.007*	↓ 0.762	↑ 0.001*	↓ 0.000*
0 – 2	↓ 0.079	↑ 0.313	↑ 0.076	↓ 0.006*

Table 1. Results of t-test — split mode (left table) and 10-fold cross validation (right table)

levels are statistically different from each other. The results of the t-test are presented in Table 1.

As it can be seen from the results of the experiments, the C4.5 and C4.5 Rules based on the split mode and C4.5 Rules and Bayes Network based on the 10-fold cross validation are significantly improved in 1-level abstraction with respect to the 0-level. On the other hand, the accuracy value of the Naive Bayes is decreased.

When we classified the 2-level abstraction represented documents, the Naive Bayes model based on the 10-fold cross validation has improved, while accuracy of the C4.5 and Bayes Network models have decreased.

Algorithms performances at 0-level abstraction with respect to 2-level are not significantly different, except decrease of accuracy of the Bayes Network model.

We explain the accuracy decreasing of most models in case of 2-level abstraction by losing some specific information in more general representation of documents. The "strange" behaviour of Naive Bayes model the in 10-fold cross validation mode, by our opinion, is justified by its specific constraints: first, it confirms independence of variables, and, second, it builds model based on all available features, while decision tree is using a feature selection procedure.

As we all know, the size of the training set affects the classification model accuracy. We believe that given a larger training set (currently in preparation) we can get more accurate results.

5 Conclusions and Future Work

In this paper we presented a new ontology-based methodology for classification of web documents to main categories according to the user "Information Needs". The main contribution of this work is using domain-based Multi-Lingual Ontology in the conceptual representation of documents. We tested our method on the specific chemicals domain, where the synonyms and the taxonomic relationships were handled. Despite the small training set, quite good results were obtained. We intend to improve current results by increasing the training set and the set of keyphrases as well as by enhancing our methodology in the following ways:

- Learning a multi-lingual domain ontology exploiting machine learning techniques.
- Elaborating (or use some existing tools like GATE [17]) for automatic construction of ontologies on specific domain. Such update will enable us to make an ontology-based classification system completely domain-independent.
- Using several ontologies for the same set of documents (or one ontology including several hierarchies).
- Mapping web documents into Web hierarchy (it may be topic ontology) to improve the classification accuracy.

Acknowledgement. We wish to thank D. Berenstein, the domain expert, for helping us in the ontology construction and collection of the training set for the learning algorithms.

References

1. R. Kosala and H. Blockeel. Web mining research: a survey. SIG KDD Explorations, Vol. 2, pp. 1-15, July 2000.
2. O. Etzioni. The World Wide Web: Quagmire or Gold Mine? Communications of the ACM, Vol. 39, No. 11, pp. 65-68. Nov. 1996.
3. M. Montes-y-Gomez, A. Gelbukh and A. Lopez-Lopez. Mining the News: Trends, Associations, and Deviations. Computacion y Sistemas, Vol. 5 No. 1, pp. 14-24, Julio-Septiembre 2001.
4. Y.Y. Yao, N. Zhong, J. Liu and S. Ohsuga, Web Intelligence (WI): research challenges and trends in the new information age, in Zhong et al., eds., Web Intelligence: research and development, LNAI 2198, Springer-Verlag, pp. 1-17, 2001.
5. R. Agrawal, T. Imielinski and A. Swami, Database mining: a performance perspective, IEEE Transactions on Knowledge and Data Engineering, Vol. 5, no. 6, pp. 914-925, 1993.
6. M. S. Chen, J. Han, and P. S. Yu, Data mining: an overview from a database perspective, IEEE Transactions on Knowledge and Data Engineering, Vol. 8(, no. 6, pp. 866-883, 1996.
7. A. Mendez-Torreblanca, M. Montes-y-Gomez and A. Lopez-Lopez. A Trend Discovery System for Dynamic Web Content Mining, [citeseer.ist.psu.edu/695212.html], 2002.
8. Tingshao Zhu, Russ Greiner and Gerald Houbl. Predicting Web Information Content. Workshop on Intelligent Techniques for Web Personalization (ITWP '03), 2003.

9. Tingshao Zhu, Russ Greiner, and Gerald Haubl. An effective complete-web recommender system. In The Twelfth International World Wide Web Conference(WWW2003), Budapest, Hungary, May 2003.
10. T. R. Gruber. A translation approach to portable ontologies. Knowledge Acquisition, Vol. 5, no. 2, pp. 199-220, 1993.
11. G. van Heijst, A.Th. Schreiber, and B.J. Wielinga. Using explicit ontologies in KBs development. IJHCS, pp. 183-291, 1997.
12. D. Billsus and M. Pazzani. A hybrid user model for news story classification. Proc. of the Seventh International Conference on User Modeling (UM '99), Banff, Canada, 1999.
13. Richard Duda and Peter Hart. Pattern Classification and Scene Analysis. Wiley, New York, 1973.
14. Andrew Jennings and Hideyuki Higuchi. A user model neural network for a personal news service. User Modeling and User-Adapted Interaction, Vol. 3, no. 1, pp. 1-25, 1993.
15. Corin R. Anderson and Eric Horvitz. Web montage: A dynamic personalized start page. Proc. of the 11th World Wide Web Conference, 2002.
16. C. Cesarano, A. d'Acerno, A. Picariello. An Intelligent Search Agent System for Semantic Information Retrieval on the Internet. Proc. of the Fifth ACM International Workshop on the Web Information and Data Management, November 7-8, 2003, New Orleans, Louisiana, USA, pp. 111-117, 2003.
17. GATE - General Architecture for Text Engineering, The Natural Language Processing Research Group, Department of Computer Science, University of Sheffield [http://gate.ac.uk/].
18. Weka - Data Mining Software in Java [http://www.cs.waikato.ac.nz/ml/weka/].
19. Li, Yuefeng and Zhong, Ning. Web Mining Model and Its Applications for Information Gathering. Knowledge-Based Systems, Vol. 17, no. 5-6, pp. 207-217, 2004.
20. Koller, D., Sahami, M. Hierarhically classifying documents using very few words. Proc. of ICML 1997.
21. Mladenic, D., Grobelnik, M. Mapping documents onto web page ontology. Web mining: from web to semantic web: EWMF 2003, Springer Lecture Notes 2004.
22. McCallum, A. et al. Improving text classification by shrinkage in a hierarchy of classes. Proc. of ICML 1998.
23. Chakrabarti, S., Dom, B., Agrawal, R., Raghavan, P. Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. The VLDB Journal (1998), Spinger-Verlag 1998.
24. Andreas Hotho, Steffen Staab, Gerd Stumme: Ontologies Improve Text Document Clustering. ICDM 2003.
25. Stephan Bloehdorn, Andreas Hotho. Text classification by boosting weak learners based on terms and concepts. Proc. of the Fourth IEEE International Conference on Data Mining, 331-334. IEEE Computer Society Press, NOV 2004.
26. Information Mapping Project. [http://infomap.stanford.edu/index.html#papers]

Ontology-Based Explanation of Discovered Associations in the Domain of Social Reality

Vojtěch Svátek[1], Jan Rauch[1] and Miroslav Flek[2]

[1] Department of Information and Knowledge Engineering,
University of Economics, Prague, W. Churchill Sq. 4, 130 67 Praha 3, Czech Republic
e-mail: svatek@vse.cz, rauch@vse.cz
[2] University College of International and Public Relations, Prague, U Santošky 17,
150 00 Praha 5, Czech Republic
e-mail: flek@vip-vs.cz

Abstract. One of possible uses of domain ontology as prior knowledge in KDD is the generation of explanations for discovered hypotheses. We developed an ontology covering a subset of concepts and relations relevant for 'municipal social reality', and manually mapped these entities on the structure of sociological data; the data originated from a comprehensive opinion poll over citizens of the capital city of Prague. The LISp-Miner KDD tool was then applied on data, and the most conspicuous associations were matched with the ontology. Some of the extracted ontology structures seem to offer useful insight into the background of empirical associations, as high-level templates that can be instantiated to concrete explanations.

1 Introduction

Domain ontologies, being hot topic in today's knowledge engineering research, are promising candidates for background knowledge to be used in the KDD process. They express the main concepts and relationships in a domain in a way that is consensual and comprehensible to the given professional community. The research in applied ontology and in KDD are, to some extent, two sides of the same coin. Ontologies describe the 'state-of-affairs' in a certain domain at an abstract level, and thus enable to verify the correctness of existing (concrete) facts as well as to infer new facts. On the other hand, KDD typically proceeds in the opposite direction: from concrete, instance-level patterns to more abstract ones. Semantic web mining [3] represents the junction of ontology and KDD research in their 'concrete' (instance-centric) corners. On the other hand, in this paper, we rather focus on the junction of 'abstract' corners, namely, of abstract ontologies themselves and general hypotheses produced by KDD.

One of the main outcomes of the first *Workshop on Knowledge Discovery and Ontologies* [5] was that the role of prior knowledge is underestimated by the KDD community, and even if this knowledge is used, it is rarely underpinned by a clear conceptual model. However, [6] demonstrated that ontologies can be beneficial in nearly all phases of the KDD (more specifically, association mining)

cycle, starting from domain and data understanding, through the semantic interpretation of discovered hypotheses, and ending by exposing the hypotheses on the semantic web, e.g. in the form of annotated textual reports [9]. Here we pay attention to the middle phase, in which the ontology is to provide 'templates' for the human expert who attempts to interpret the discovered knowledge.

The paper is structured as follows. Section 2 describes the process of designing our ontology of social reality, in a bottom-up manner. Section 3 recalls the basic principles of the LISp-Miner system which was used as knowledge discovery tool. Section 4 presents the actual experiments with using the ontology as prior knowledge for (further) knowledge discovery. Finally, section 5 reviews some related work, and section 6 shows directions for future research.

2 Designing and Mapping the Ontology

2.1 State of the Art in Social Ontology Modelling

The society as such has mostly been subject of ontology research at the philosophical level. Probably the best known recent example is the work by Searle[3]. Some notions of social reality also appeared in formal ontological engineering, for example, in the 'social' fragment of the DOLCE upper-level ontology[4], which contains concepts such as 'social relationship' or 'social institution'. Similarly, Boella & van der Torre [4] recently developed an upper-level model of social reality centred around the concept of 'agent'. In a bottom-up manner, on the other hand, a tiny fragment of social reality (namely, the relationships among and the most imminent attributes of persons) has been studied by the FOAF community, see e.g. [10]. What we however needed in our project was a comprehensive formal model spanning across many heterogeneous areas; we therefore decided to create a new ontology, in a *bottom-up* manner.

2.2 Designing the Ontology

Both the *ontology* and the *dataset* used for association discovery had the same seed material: the questionnaire[5] posed to respondents during the *opinion poll* mapping the 'social climate' of the capital city of Prague in Spring 2004. The questionnaire contained 51 questions related to e.g. economic situation of families, ways of earning money and dwelling, or attitude towards important local events, political parties or media. Some questions consisted of aggregated sub-questions each corresponding to a different 'sign', e.g. "How important is X for you?", where X stands for family, politics, religion etc. Other questions corresponded each to a single 'sign'. While the *dataset* was straightforwardly

[3] See [15] for a summary.

[4] `http://dolce.semanticweb.org/`

[5] The questionnaire was designed by sociology experts, entirely independent of the KDD and ontological engineering research described in this paper.

derived from the individual 'signs', each becoming a database column[6], the *ontology* first had the form of *glossary* of candidate terms (manually) picked from the text of the questions; duplicities were removed. In conformance with most ontology engineering methodologies [8], the terms were then divided into candidates for *classes*, *relations* and *instances*, respectively. Then a *taxonomy* and a structure of *non-taxonomic relations* was (again, manually) built, while filling additional entities when needed for better connectivity of the model or just declared as important by domain expert. The instances either corresponded to enumerated values of properties (modelled according to the W3C note [14]), e.g. `GOOD_JOB_AVAILABILITY`, or to outstanding individuals such as `PRAGUE` or `CHRISTIAN_DEMOCRATIC_PARTY`, as these were often referred to in the text of the questionnaire.

The current version of the ontology, eventually formalised in OWL[7], consists of approx. 100 classes, 40 relations and 50 individuals[8]. A Protégé[9] window showing parts of the class hierarchy plus the properties of class `Person` is at Fig. 1. Note that the ambition of our ontology is not to become a widely-usable formal model of social reality; it rather serves for 'simulation' of the possible role of such ontology in the context of KDD. More details on the process of designing the ontology (in particular, the 'design patterns' used) can be found in [17].

2.3 Data-to-Ontology Mapping

The second and somewhat easier part of the knowledge engineering phase of our project was to *map* the attributes of the dataset to ontology concepts, relations and instances. Since the core of the ontology had been manually designed based on the text of the questions, it sufficed to track down the links created while building the ontology and maintained during the concept-merging phase. An example of mapping between a question and (fragments of) the ontology is in Table 1. Emphasised fragments of the text map to the concepts `Job_availability`, `Metropoly` and `Family` and to the individuals `GOOD_JOB_AVAILABILITY`, `PRAGUE`, `CENTRAL_EUROPE` and `EU`, plus several properties not shown in the diagram. Note that question no.3 is a 'single-sign' question, i.e. it is directly transformed to one data attribute used for mining. In addition to questions, ontology mapping was also determined for *values* allowed as answers, especially for questions requiring to select concrete objects (city districts, political parties etc.).

[6] And, subsequently, an attribute for the LISp-Miner system, see the next section.

[7] `http://www.w3.org/2004/OWL`

[8] By naming convention we adopted, individuals are in capitals, classes start with capital letter (underscore replaces inter-word space for both individuals and classes), and properties start with small letter and the beginning of other than first word is indicated by a capital letter.

[9] `http://protege.stanford.edu`

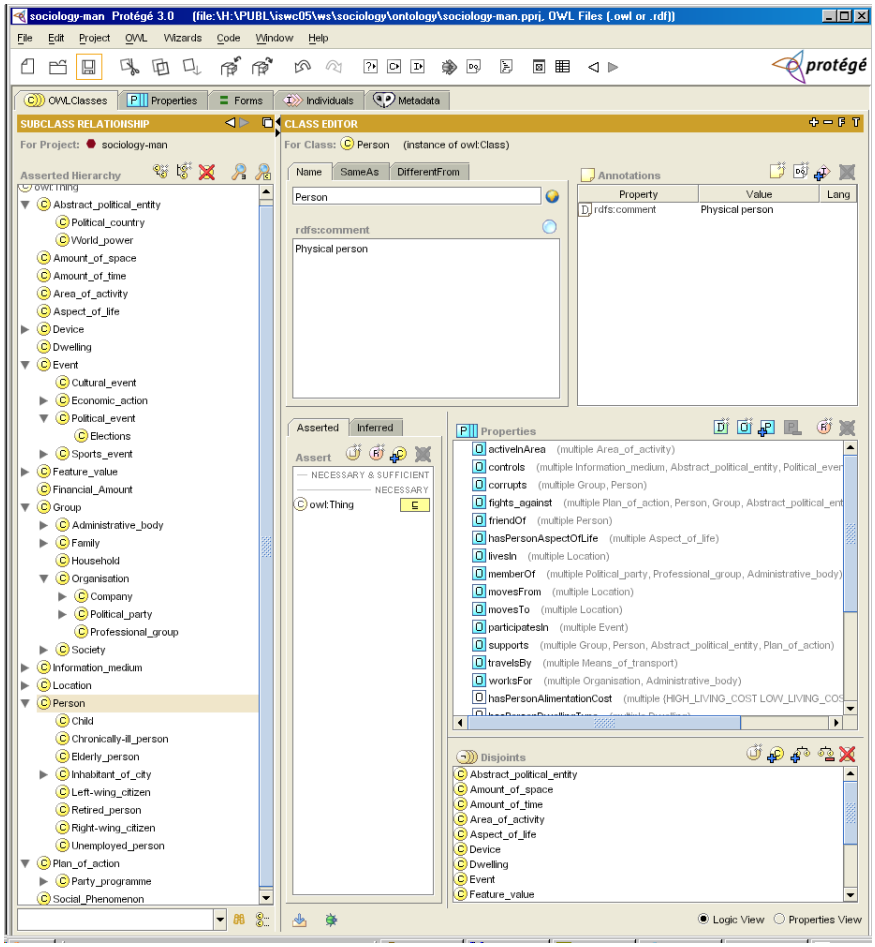


Fig. 1. Incomplete view of the ontology in Protégé

3 Association Mining with 4ft-Miner

The *4ft-Miner* procedure is the most frequently used procedure of the *LISp-Miner* data mining system [13]. It mines for association rules of the form $\varphi \approx \psi$, where φ and ψ are called *antecedent* and *succedent*, respectively[10]. Antecedent and succedent are conjunctions of *literals*. Literal is a Boolean variable $A(\alpha)$ or its negation $\neg A(\alpha)$, where A is an *attribute* (corresponding to a column in the data table) and α (a set) is *coefficient* of the literal $A(\alpha)$. The literal $A(\alpha)$ is true for a particular object o in data if the value of A for o is some v such that $v \in \alpha$.

[10] *4ft-Miner* also mines for *conditional* hypotheses (i.e. with a third symbol representing a restrictive condition). We will not discuss them here, for brevity.

From May 1, 2004, *Prague* will become one of *Central-European metropolies* of the *EU*. Do you think that this fact will improve the *availability of jobs* for you or for your *relatives*?

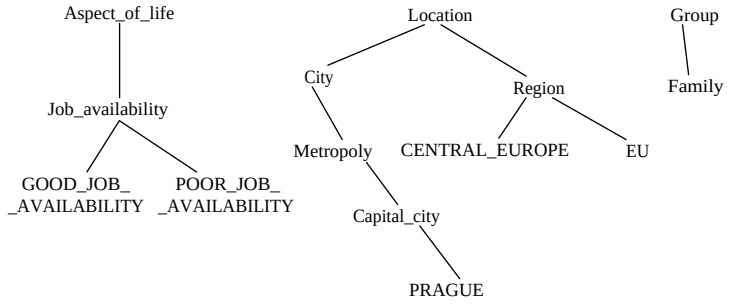


Table 1. Question no.3 and fragments of ontology used for its mapping

The association rule $\varphi \approx \psi$ means that the Boolean variables φ and ψ are associated in the way defined by the symbol $\approx$. The symbol $\approx$ is called *4ft-quantifier*. It corresponds to a condition over the four-fold contingency table of φ and ψ. The four-fold contingency table of φ and ψ in data matrix $\mathcal{M}$ is a quadruple $\langle a, b, c, d\rangle$ of natural numbers such that a is the number of data objects from $\mathcal{M}$ satisfying both φ and ψ, b is the number of data objects from $\mathcal{M}$ satisfying φ and not satisfying ψ, c is the number of data objects from $\mathcal{M}$ not satisfying φ and satisfying ψ, and d is the number of from $\mathcal{M}$ from $\mathcal{M}$ satisfying neither φ nor ψ.

There are 16 4ft-quantifiers in the 4ft-Miner. An example of 4ft-quantifier is *above-average dependence* ,
$\sim^{+}_{p,Base}$, which is defined for $0 < p$ and $Base > 0$ by the condition

$$\frac{a}{a+b} \geq (1+p)\frac{a+c}{a+b+c+d} \wedge a \geq Base \ .$$

The association rule $\varphi \sim^{+}_{p,Base} \psi$ means that among the objects satisfying φ is at least $100p$ per cent more objects satisfying ψ than among all observed objects and that there are at least $Base$ observed objects satisfying both φ and ψ.

As an example of association rule, let us present the expression

$$A(a_1, a_7) \wedge B(b_2, b_5, b_9) \sim^{+}_{p,Base} C(c_4) \wedge \neg D(d_3)$$

Here, $A(a_1, a_7)$, $B(b_2, b_5, b_9)$, $C(c_4)$ and $\neg D(d_3)$ are literals, a_1 and a_7 are categories of A, and $\{a_1, a_7\}$ is the coefficient of $A(a_1, a_7)$[11], and analogously for the remaining literals.

[11] For convenience, we can write $A(a_1, a_7)$ instead of $A(\{a_1, a_7\})$.

Note that the hypothesis definition language of 4ft-Miner is far richer than we described. For the sake of this paper, the description above is sufficient; for more information see e.g. the project homepage `http://lispminer.vse.cz` or [13].

4 Experiments

4.1 Overview

We experimented with various 4ft-Miner settings on the poll dataset, mostly using the *above-average dependence* quantifier explained in previous section. As we did not want to restrict the choice of antecedent and succedent of hypotheses, between which the chains of ontology entities were to be found, we kept the task definition maximally general: any of 96 attributes (corresponding to 'signs' from the questionnaire) was allowed in antecedent as well as in succedent. As we wanted to start with (structurally) simplest possible hypotheses, we set the length of antecedent as well as of succedent to 1, and the cardinality of coefficient also to 1 (i.e., choice of single value of the attribute). The run-times were typically lower than a second.

We divided the strong hypotheses resulting from 4ft-Miner runs into four groups, with respect to their amenability to *ontology-based explanation*:

1. *Strict logical dependencies*, an example of which is the association between answers to the questions "Do you use a public means of transport?" and "Which public means of transport do you use?". Such hypotheses are of no interest as KDD results in general.
2. Relationships amounting to *obvious causalities*, for example, the association between "Are you satisfied with the location where you live?" and "Do you intend to move?" Such relationships (in particular, their strengh) might be of some interest for KDD in general; however, there is no room for ontology-based explanation, since both the antecedent and succedent are mapped on the same or directly connected ontology concepts (`Location`, `livesIn`, `movesFrom` etc.).
3. Relationships between signs that have the character of respondent's agreement with relatively *vague propositions*, for example "Our society changes too fast for a man to follow." and "Nobody knows what direction the society is taking." We could think of some complex ontology relationships, however, by Occam's razor, it is natural just to assume that the explanation link between the antecedent and succedent goes through the categorisation of the respondent as conservative/progressist or the like.
4. Relationships between signs corresponding to concrete and relatively *semantically distant* questions (namely, appearing in different question 'groups' or 'clusters'). This might be e.g. the question "Do you expect that the standard of living of most people in the country will grow?", with answer 'certainly not', and the question "Which among the parties represented in the city council has a programme that is most beneficial for Prague?" with 'KSČM'

(the Czech Communist Party) as answer. Such *cross-group* hypotheses are often amenable to ontology-based explanation. We'll elaborate on this particular example in the following subsection.

Since we do not (yet) have an appropriate software support for extracting entity chains (i.e. explanation templates) from the ontology, we examined it via manual browsing. As a side-effect of chain extraction, we also identified *missing* (though obvious) links among the classes, which could be added to the ontology, and also some modelling *errors*, especially, domain/range constraints at an inappropriate level of generality.

4.2 Example of Explanation Template Set

The hypothesis from the last example above, formally written as `Z05(4)` $\sim^{+}_{0.22,64}$ `Z18(3)`, could be visualised by the available means of LISp-Miner as shown at Fig. 2 and Fig. 3.

The first view presents the *four-fold contingency table*:

- 64 people disagree that the standard of living would grow AND prefer KSČM
- 224 people disagree that the standard of living would grow AND DO NOT prefer KSČM
- 171 people DO NOT disagree[12] that the standard of living would grow AND prefer KSČM
- 2213 people DO NOT disagree that the standard of living would grow AND DO NOT prefer KSČM.

The contingency table is followed with a long list of computed characteristics.

The second view presents the same information *graphically*. We can see that among the people who disagree that the standard of living would grow, there is a 'substantially' higher number of people who also prefer KSČM than in the whole data sample, and vice versa[13]. The whole effort of formulating hypotheses about the reason for this association is however on the shoulders of the human expert.

In order to identify potential *explanation templates*, we took advantage of the *mapping* created prior to the knowledge discovery phase, see section 2.3. The negative answer to the question about standard of living was mapped to the individual `BAD_LIVING_STANDARD` (instance of `Social_phenomenon`), and the respective answer to the question about political parties was mapped to the class `Political_party`, to its instance `KSCM`, to the class `Party_programme` and to the class `City_council`.

There are many ways of *ordering* the explanation templates; here we order them first by the decreasing number of involved entities on which the hypothesis is *mapped* and then by the decreasing number of *all* involved entities. The templates do not contain intermediate classes from the hierarchy (which are not even

[12] More precisely, their answer to the question above was not 'certainly not'; it was one of 'certainly yes', 'probably yes', 'probably no'.

[13] This is the principle of the *above-average* quantifier, which is symmetrical.

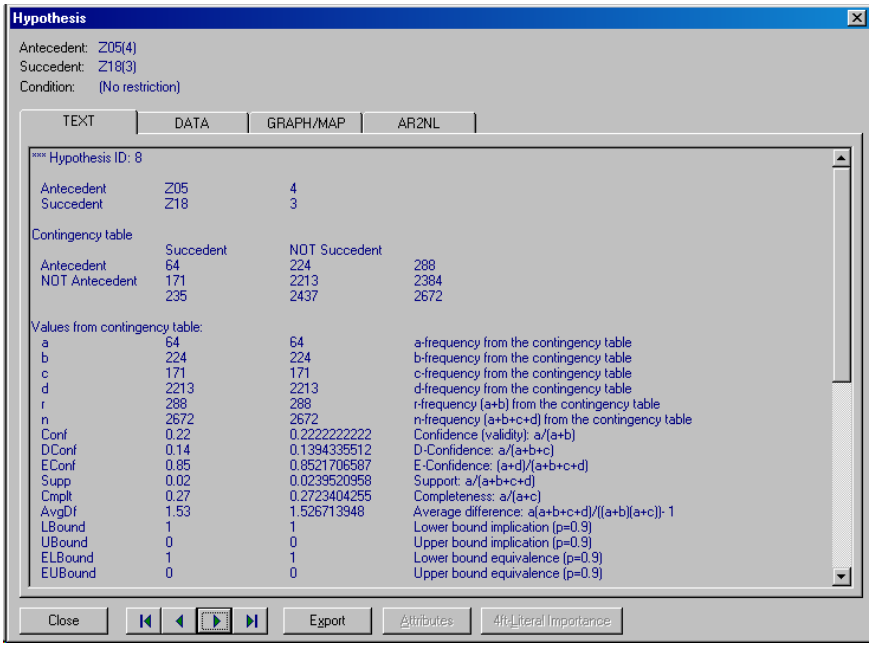


Fig. 2. Textual view of a LISp-Miner hypothesis

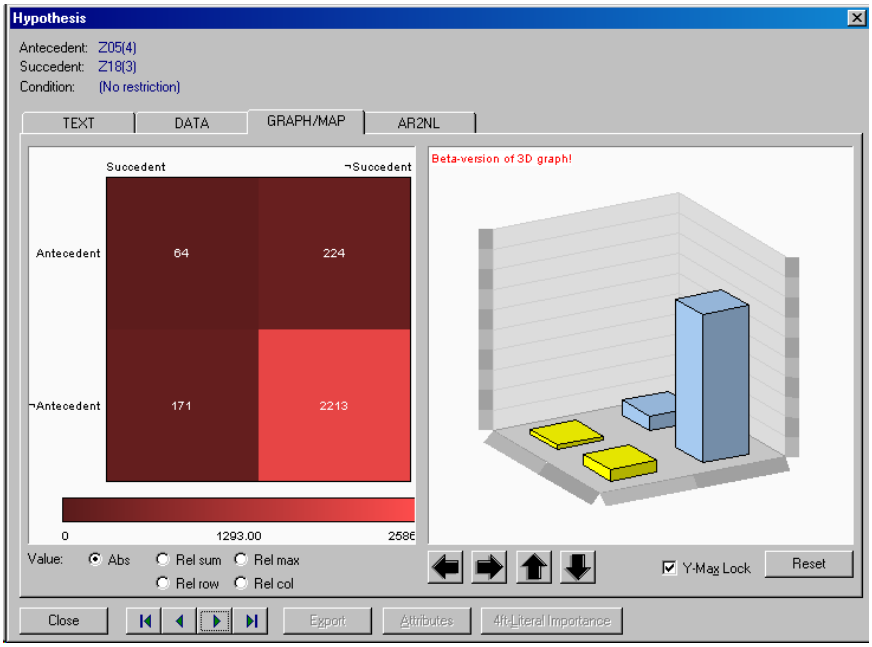


Fig. 3. Graph-based view of a LISp-Miner hypothesis

Template	Mapped	All
KSCM ∈ **Political_party** hasPartyProgramme **Party_programme** ⊑ Plan_of_action hasObjective Social_phenomenon ∋ **BAD_LIVING_STANDARD**	4	6
KSCM ∈ **Political_party** isRepresentedIn Administrative_body ⊒ **City_council** carriesOutAction Economic_action hasImpactOn Social_phenomenon ∋ **BAD_LIVING_STANDARD**	4	7
KSCM ∈ **Political_party** hasPartyProgramme **Party_programme** ⊑ Plan_of_action envisagesAction Action ⊒ Economic_action hasImpactOn Social_phenomenon ∋ **BAD_LIVING_STANDARD**	4	8
KSCM ∈ Group informsAbout Social_phenomenon ∋ **BAD_LIVING_STANDARD**	2	3
KSCM ∈ Group carriesOut Action ⊒ Economic_action hasImpactOn Social_phenomenon ∋ **BAD_LIVING_STANDARD**	2	6
KSCM ∈ Group participatesIn Event ⊒ Economic_action hasImpactOn Social_phenomenon ∋ **BAD_LIVING_STANDARD**	2	6
KSCM ∈ Group supports Action ⊒ Economic_action hasImpactOn Social_phenomenon ∋ **BAD_LIVING_STANDARD**	2	6
KSCM ∈ Group fightsAgainst Group carriesOutAction Action ⊒ Economic_action hasImpactOn Social_phenomenon ∋ **BAD_LIVING_STANDARD**	2	7

Table 2. Explanation templates for 'standard of living' vs. 'KSČM' association

counted for the ordering). Relations, i.e. OWL properties, are only considered as linked to the class for which they are directly defined as domain/range, i.e. not to the class that just inherits them. Table 2 lists some (by far not all) possible templates, with the counts of *mapped* and *all* entities of which the template consists, respectively. The symbols ⊑, ⊒ stand for subclass/superclass relationship and ∈, ∋ for instance-to-class membership[14].

We can see that the 'most preferable' template suggests that the KSČM party may have some programme that may have as objective to reach the phenomenon of BAD_LIVING_STANDARD. The second looks a bit more adequate: the KSČM party is represented in the city council that can carry out an economic action that may have some impact on the phenomenon of BAD_LIVING_STANDARD. The third is almost identical to the first one. The fourth (and simplest) might actually be most plausible: the KSČM party informs about the phenomenon of

[14] Note that this description-logic-like notation is only used here for brevity; a more user-oriented (e.g. graphical) representation would probably be needed to provide useful support for a domain expert not familiar with knowledge representation conventions.

BAD_LIVING_STANDARD. Let us finally mention the fifth template, which builds on an incorrect 'inference' (caused by imprecise modelling): the party is assumed to carry out an economic action, which it (directly) can't. The relation was defined with Group and Action as subsets of its domain and range, respectively. However, the combination of Political_party (subclass of Group) and Economic_action (subclass of Action) is illegal and should have been ruled out by an axiom such as Political_party $\sqsupseteq$ (ALL carriesOutAction (NOT Economic_action)).

5 Related Work

Although domain ontologies are a popular instrument in many diverse applications, they only scarcely appeared in 'tabular' KDD, so far. A notable exception was the work by Philips & Buchanan [12], where 'common-sense' ontologies of time and processes were exploited to derive constraints on attributes, which were in turn used to construct new attributes. Although not explicitly talking about ontologies, the work by Clark & Matwin [7] is also relevant; they used qualitative models as bias for inductive learning. Finally, Thomas et al. [18] and van Dompseler & van Someren [19] used problem-solving method descriptions (a kind of 'method ontologies') for the same purpose. There have also been several efforts to employ taxonomies over domains of individual attributes [1, 2, 11, 16] to guide inductive learning. None of these projects however attempted to explore the role of domain ontology in *interpreting* the results of the mining process.

For a brief review of related work on *social ontology modelling* proper see section 2.1.

6 Conclusions and Future Work

We described a simple experiment in matching a social reality ontology to hypotheses discovered via data mining from poll data; abstract templates for possible explanations of the hypotheses were identified.

The work is only in its early phase, as our ontology reflects the state of affairs in our 'domain' in a very imprecise and simplified way[15]. Its further extension and refinement in close contact with the expert is envisaged; we also plan to take into account prior work in (philosophical as well as applied) social reality modelling mentioned in section 2.1. We would also like to pay more attention to expressing (mainly as relation instances) *additional heuristic knowledge* available in our domain, which could help automatically fill the templates with concrete relationships. Such a (not yet formalised) knowledge base actually arose in connection with the polls in question.

With growing body of available knowledge, *end-user tests* would become more meaningful. An important step would be to proceed from the current,

[15] An important problem, which is however not easy to overcome by state-of-the-art ontology engineering technology, is the static character of our model.

subjective, evaluation of patterns to quantitative evaluation of their efficiency in supporting the interpretation of hypotheses.

Furthermore, we would like to follow up with our earlier effort to expose KDD results on the *semantic web* [9]. Aside 'plain' empirical hypotheses, instantiated explanation templates endorsed by an expert could straightforwardly be represented.

From the point of view of *association discovery*, the experiments revealed the utility of further extensions to the task definition principles of LISp-Miner, in particular regarding the search for *cross-group* hypotheses. Such extensions would make further experiments with ontologies or similar background models more efficient.

In a longer run, it would also be desirable to extend the scope of the project towards discovered hypotheses with *more complex structure*, e.g. with longer antecedents/succedents, with additional condition, or even to hypotheses discovered by means of a different procedure. An example of the last is the recently implemented procedure SD4FT; it searches for pairs of sets of objects in data such that one appears in different empirical associations than the other. Generating explanations for such hypotheses would be much more demanding but would also provide greater benefits.

Acknowledgements

The research is partially supported by the grant no.201/05/0325 of the Czech Science Foundation, "New methods and tools for knowledge discovery in databases".

References

1. Almuallim, H., Akiba, Y. A., Kaneda, S.: On Handling Tree-Structured Attributes in Decision Tree Learning. In: Proceedings of the Twelfth International Conference on Machine Learning (ML-95). Morgan Kaufmann, 12–20.
2. Aronis, J.M., Provost, F.J., Buchanan, B.G.: Exploiting Background Knowledge in Automated Discovery. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, 1996 (KDD-96).
3. Berendt, B., Hotho, A., Stumme, G.: 2nd Workshop on Semantic Web Mining, held with at ECML/PKDD-2002, Helsinki 2002, `http://km.aifb.uni-karlsruhe.de/semwebmine2002`.
4. Boella, G., van der Torre, L.: An Agent-Oriented Ontology of Social Reality. In: Proc. FOIS'04, Torino 2004, Springer LNCS.
5. Buitelaar, P., Franke, J., Grobelnik, M., Paass, G., Svátek, V. (eds.): ECML/PKDD 2004 Workshop on Knowledge Discovery and Ontologies (KDO-04), Pisa 2004.
6. Češpivová, H., Rauch, J., Svátek V., Kejkula M., Tomečková M.: Roles of Medical Ontology in Association Mining CRISP-DM Cycle. In: ECML/PKDD04 Workshop on Knowledge Discovery and Ontologies (KDO'04), Pisa 2004.
7. Clark, P. Matwin, S.: Using Qualitative Models to Guide Inductive Learning. In: Machine Learning - ECML'94, European Conference on Machine Learning, Catania 1994. Lecture Notes on Artificial Intelligence, Springer Verlag 1994, 360–365.

8. Gómez-Perez, A., Fernández-Lopez, M., Corcho, O.: Ontological Engineering. Springer 2004.
9. Lín, V., Rauch, J., Svátek, V.: Content-based Retrieval of Analytic Reports. In: Schroeder, M., Wagner, G. (eds.). Rule Markup Languages for Business Rules on the Semantic Web, Sardinia 2002, 219–224.
10. Matsuo, Y., Hamasaki, M., Mori, J., Takeda, H., Hasida, K.: Ontological Consideration on Human Relationship Vocabulary for FOAF. In: 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web, Galway 2004.
11. Núñez, M.: The Use of Background Knowledge in Decision Tree Induction. *Machine Learning*, 6, 231–250 (1991).
12. Phillips, J., Buchanan, B.G.: Ontology-guided knowledge discovery in databases. In: International Conf. Knowledge Capture, Victoria, Canada, 2001.
13. Rauch, J., Šimůnek, M.: An Alternative Approach to Mining Association Rules. In: Lin, T. Y., Ohsuga, S., Liau, C. J., and Tsumoto, S. (eds.), Data Mining: Foundations, Methods, and Applications, Springer-Verlag, 2005, pp. 219–238 (*to appear*)
14. Rector, A. (ed.): Representing Specified Values in OWL: "value partitions" and "value sets". W3C Working Group Note, 17 May 2005, online at `http://www.w3.org/TR/swbp-specified-values/`.
15. Searle, J. R.: Social Ontology: Some Basic Principles. Available online from `http://ist-socrates.berkeley.edu/~jsearle/articles.html`.
16. Svátek, V.: Exploiting Value Hierarchies in Rule Learning. In: van Someren, M. - Widmer, G. (Eds.): ECML'97, 9th European Conference on Machine Learning. Poster Papers. Prague 1997, 108–117.
17. Svátek, V.: Observations from Development of Social Reality Ontology for a KDD Application. Submitted paper.
18. Thomas J., Laublet, P., Ganascia, J. G.: A Machine Learning Tool Designed for a Model-Based Knowledge Acquisition Approach. In: EKAW-93, European Knowledge Acquisition Workshop, Lecture Notes in Artificial Intelligence No.723, N.Aussenac et al. (eds.), Springer-Verlag, 1993, 123–138.
19. van Dompseler, H. J. H., van Someren, M. W.: Using Models of Problem Solving as Bias in Automated Knowledge Acquisition. In: ECAI'94 - European Conference on Artificial Intelligence, Amsterdam 1994, 503–507.

Author Index