STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LIX**, Special Issue 2, 2014
ICFCA 2014: 12th International Conference on Formal Concept Analysis, Cluj-Napoca, June 10-13, 2014

STABILITY-BASED FILTERING FOR ONTOLOGY RESTRUCTURING

SCHAHRAZED FENNOUH, ROGER NKAMBOU, PETKO VALTCHEV, AND MOHAMED ROUANE-HACENE

Abstract. Assessing the relevance of concepts extracted from data is an important step in the knowledge discovery process. We address this issue in a specific outfit, i.e., the discovery of new ontological abstractions by relational concept analysis (RCA). In the context of RCA-based ontology restructuring, potentially relevant abstractions must be recognized among the formal concepts of the output lattice before integrating them into the restructured ontology. Thus, a key technical challenge is the design of effective relevance-based filtering methods. In our study, we examined a variety of relevance measures. Here, we focus on concept stability and discuss its usefulness in the light of the outcome from an experimental study involving several ontologies retrieved from the Web.

1. Introduction

An ontological model is like a database conceptual schema [8]: it provides the framework in which to fit the fine-grain knowledge about a particular domain or subject. Like most artifacts in information system development (conceptual models, design models, source code, etc.), an ontological model is prone to errors and design anomalies. Previous attempts at detecting and, possibly, correcting such anomalies yielded a variety of *restructuring* approaches. Intuitively, a restructuring process aims at improving the quality of an ontology, which further increases usability and eases maintenance [3]. Technically speaking, ontology restructuring reshuffles its current structure into a new one, better organized and more complete. It thus refers to: (1) correction and reorganization of knowledge contained in the initial conceptual model, and (2) the discovery of missing knowledge pieces and their integration into the improved structure [19].

Received by the editors: March 27, 2014.
2010 *Mathematics Subject Classification.* 03G10, 68T30.
1998 *CR Categories and Descriptors.* I.2.4 [**K**nowledge Representation Formalisms and Methods]: Representations.
Key words and phrases. Filtering, Relevance evaluation, Concept stability, Ontology restructuring.

The problem has been addressed in the literature from a variety of standpoints [3, 9, 20]. However, there is no such thing as a well-established methodology covering the variety of restructuring steps and techniques. Even worse, none of the proposed solutions offers a holistic approach to the ontological structure. Instead, local changes are focused on, without insight on their impact on distant parts of the structure. In addition, existing methods are limited to problem detection and improvement, leaving the other crucial restructuring aspect, i.e., the discovery of missing knowledge chunks, uncovered.

Following the success of FCA-based restructuring methods in software re-engineering [6], we propose a similar approach for ontologies. Indeed, FCA provides the formal framework necessary to support a truly holistic approach towards restructuring while simultaneously propping up new abstractions through factoring out shared descriptions. Moreover, due to the complex relational information comprised in a typical ontological model, we propose to use the Relational Concept Analysis (RCA) extension. Yet the mathematical strength of FCA and the expressiveness of RCA come with a price: A key challenge to face here is the complexity of the resulting relational lattices. A standard way out in such cases is the design of effective filtering methods that help spot and then remove the spurious concepts that abound in the output lattice. Thus, our overall goal is the design of appropriate decision criteria for relational concepts or, alternatively, means to assess their relevance.

Selecting relevant concepts within a lattice is knowingly a delicate task for which few generic hints are available. Indeed, relevance is a contextual and subjective property. Therefore, fully automated filtering methods rely on structural properties that are easier to measure. In our restructuring context, however, the input ontology, albeit of a flawed structure, constitutes a rich source of domain knowledge to explore in the design of "semantic" relevance measures. Yet at this stage, we chose to keep to a purely structural approach and ignore the ontology. Thus, we adapted concept *stability* as defined in [10] to our RCA framework. The present paper is a report on an experimental evaluation of the resulting measure's usefulness.

The remainder of the paper is organized as follows; we start, in section 2, by presenting our RCA-based approach for ontology restructuring; we recall the basic notions of FCA/RCA framework and present the problem of lattice filtering. Section 3 is devoted to the definition of the notion of stability and its projection in an ontological context; and then the proposal of a simple filtering heuristic based on stability. We explain, in section 4, our experimental framework followed by our experimental study including the discussion of our results. Section 5 provides an overview of related work. Finally, section 6 provides concluding remarks with an outlook of future work.

TABLE 1. Key differences with standard FCA notations.

Not.	Description	Not.	Description
O	The set of formal objects	$\mathcal{C}_K^o$	The family of extents of a context K
A	The set of formal attributes	$\mathcal{C}_K^a$	The family of intents of K
$\mathcal{C}_K$	The set of formal concepts of K	$\mathcal{L}_K$	The concept lattice of K

Classes	author	reviewer	paper	report	name	affiliation	email	rank	title	contents	comments	login	passw
Author	×				×	×		×					
Paper			×						×	×			
Reviewer		×			×	×	×					×	×
Report				×							×		

Properties	'write'	'compose'	'rate'
Write	×		
Compose		×	
Rate			×

FIGURE 1. Contexts K_1 (of *Classes*) and K_2 (of *Object properties*) of CMS ontology.

2. Ontology Restructuring Using RCA

FCA has been successfully applied as a formal framework for the design/restructuring of class hierarchies in OO languages [4, 6] and of conceptual hierarchies in knowledge representation [18, 12]. Below, we first recall basic notions from FCA and RCA, then present the overall restructuring method and finally state the filtering problem for concept lattices output by RCA.

2.1. RCA basics.

The notations we use in the remainder of this paper slightly diverge from the standard ones. The important differences are summarized in Table 1.

The aim of RCA [17] is to discover formal concepts on top of multiple object sets described by both proper attributes and links. In RCA, data encoding is done through a structure called *Relational Context Family* (RCF). RCF is a pair **(K,R)**, such that: $\mathbf{K} = \{K_i\}_{i=1,...,n}$ a set of contexts $K_i = (O_i, A_i, I_i)$, each representing an object species, and $\mathbf{R} = \{r_k\}_{k=1,...,m}$ a set of relations r_k where $r_k \subseteq O_{i_1}$ x O_{i_2} for some $i_1, i_2 \in \{1, ..., n\}$, with O_{i_1} (domain of r_k) and O_{i_2} (range of r_k) are the object sets of the contexts K_{i_1} and K_{i_2}, respectively. Fig. 1 shows two contexts K_1 and K_2 of a *Conference Management System* (CMS) ontology representing two classes of ontological entities, *concepts* and *object properties*, respectively (see Fig. 2 for the corresponding concept lattices $\mathcal{L}_1$ and $\mathcal{L}_2$).

Existing links between *concepts* and *object properties* are represented by four relations (*source, target, domain and range*) as showed in Fig. 3. For instance, the *domain* relation models the relationship between a property and

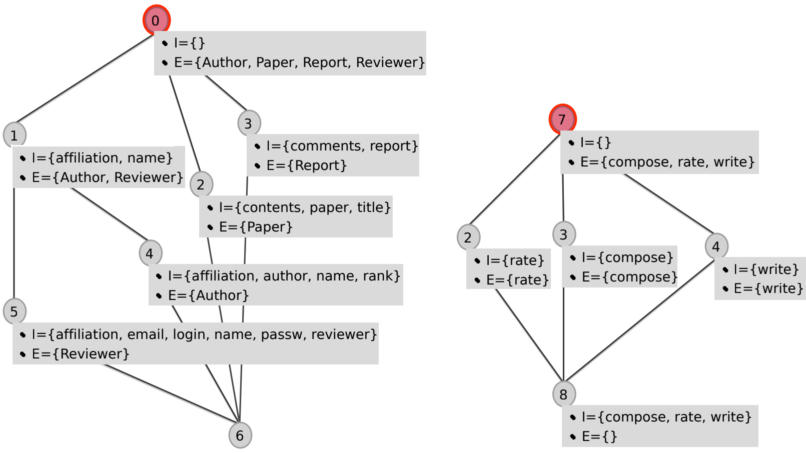

FIGURE 2. The concept lattices of the contexts K_1 (left) and K_2 (right).

source	Write	Compose	Rate
Author	×		
Paper			
Reviewer		×	
Report			×

target	Write	Compose	Rate
Author			
Paper	×		×
Reviewer			
Report		×	

domain	Author	Paper	Reviewer	Report
Write	×			
Compose			×	
Rate				×

range	Author	Paper	Reviewer	Report
Write		×		
Compose				×
Rate		×		

FIGURE 3. Relations of CMS Relational Context Family.

the own class and the *source* relation expresses the semantic of “is the domain of”.

To deal with the relational structure of an RCF, a mechanism called *scaling* transforms inter-object links into descriptors for formal objects. As a result, new attributes called *relational* are added to the attribute sets A_i from the RCF. Thus, for $K_j = (O_j, A_j, I_j) \in \mathbf{K}$, A_j is extended with attributes $a_{r,c}$ where r is a relation from $\mathbf{R}$ such that *dom(r)* $\subseteq O_j$, and c is a concept over the objects from *ran(r)*. Furthermore, such attributes involve a quantifying operator (universal, existential, existential with cardinality restriction, etc.). In our CMS case, all such attributes are assumed to refer to an existential quantifier. For instance, in Fig. 4 (left hand side), the attribute source:c5 of the concept $c_{\#1}$ is to be read as: *For each class* o *in the extent of* $c_{\#1}$*, exists an object property* p *in the extent of* $c_{\#5}$ *(*right hand side*) such that* o *is the source of* p.

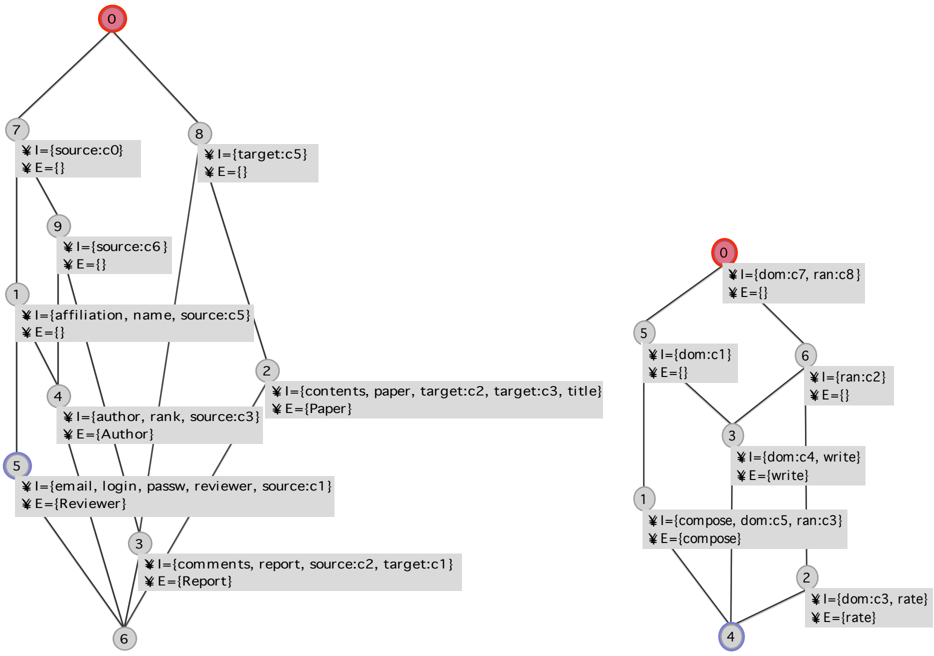

FIGURE 4. The fixed-point lattices of CMS ontology.

The overall process of RCA follows a Multi-FCA method which allows to construct a set of lattices, called *Relational Lattice Family* (RLF), one per relational context. The RLF is defined as a set of concepts that jointly reflect all attributes and links shared by the objects of the RCF. The logic of this analysis method is iterative one: Whenever the contexts of RCF are extended, their corresponding lattices expand as well. At the initialization step, each context K_i^0 is obtained from K_i by applying a conceptual scaling to the many-valued attributes in K_i and then the lattices $\mathcal{L}_i^0$ are constructed. At the step p and for each relation $r_k \subseteq O_i \times O_j$, the lattice $\mathcal{L}_j^{p-1}$ of the range context is used to extend the domain context K_i^p and then update the lattice of the domain context $\mathcal{L}_i^p$. The process ends whenever at two subsequent steps all the pairs of corresponding lattices $\mathcal{L}_i^n$ and $\mathcal{L}_i^{n+1}$ are isomorphic (i.e., the fix point solution is reached). For instance, the fixed-point lattices of the CMS example are depicted in Fig. 4 : within the property lattice (see the right of the figure), the concept $c_{\#5}$ summarizes the commonalities of write and compose which amount to a shared domain class, represented by the concept $c_{\#1}$ from the class lattice (see the left hand side of the figure), that is the super-class of Author and Reviewer.

2.2. **Restructuring Approach.** Our restructuring approach follows five steps: alignment, encoding, analysis, filtering and reverse encoding. The *alignment* step compares the elements of the initial ontology to identify similarities. This will eliminate redundancies and avoid duplication in the codes of similar ontological elements. In *encoding* step, the initial ontology will be transformed into a unique Relational Context Family (RCF) where each context represents a class of ontology metamodel elements (e.g. concepts and roles). Existing links in the metamodel between these two sorts of elements are represented by cross-context relations (e.g. source, target, domain and range). The *analysis* step consists of construction of the initial lattices (concepts and roles lattices) with FCA, then translates the cross-context relations into relational attributes following "relational scaling" mechanism. Next, the final concept lattices are constructed according to RCA iterative process converging towards a global fix-point of the analysis.

Please notice that the final lattices provide the support for the reorganization of the initial ontology. The *filtering* phase will filter these lattices by pruning of uninteresting and spurious formal concepts. The way to proceed at that stage can be summarized as follows:

(1) *identifying the ontological concepts of the domain (from the initial ontology),*
(2) *selecting the relevant new abstractions.*

The final step is the *reverse encoding* that consists to generate semi-automatically (with the participation of the expert) the restructured ontology model. The idea is to translate the formal concepts considered as relevant in the filtered lattices into ontological elements in OWL format.

2.3. **Filtering of Concept Lattices.** While providing a strong mathematical background, FCA rises a serious issue that is rooted the overly large-sized lattices which typically contain many spurious concepts. RCA tends to generate even larger lattices since it deals with richer data structures. In our context, a *formal concept* will represent an *ontological concept* (or class) where the intent and the extent correspond, respectively, to the set of the properties of the latter and the set of its sub-concepts.

In the final lattice produced by RCA, each formal concept represents a candidate likely to be selected as an ontological concept by an expert if deemed relevant (or otherwise if irrelevant). Thus, two related questions can be asked:

- *What is the ontological value of a formal concept?*
- *How to recognize among the large number of candidates the relevant ones?*

In studies from the literature, such as [18, 12] aiming at learning or merging of ontologies with FCA, there is no suggestion on an evaluation method of formal concept quality. Most of them focus on expert-based validation to create the target ontology from the final concept lattices. We tried to address this issue in our context of ontology restructuring using RCA. Our objective is to propose effective metrics to evaluate the relevance of formal concepts which will constitute the restructured ontology.

To that end, we took some inspiration from: (1) the principles and requirements that must be respected by an ontological model [7]; (2) the work on the evaluation of ontologies, including those which consider ontology as a graph and try to detect its structural and semantic characteristics [1]; and (3) metrics for lattices pruning [11, 14]. We chose four metrics inclusive two structural ones: *Density* indicates the usefulness of a concept in terms of the additional information it provides w.r.t. its neighborhood. *Stability* measures the dependency of a concept upon single objects/attributes of its. The remaining two are semantics-based: *Semantic Similarity between children concepts* reflects the semantic correctness of a concept, i.e., to what extent it subsumes similar sub-concepts. *Semantic Similarity with the user center of interest* assess the level to which the concept is rooted in the important concepts from the initial ontology (in terms of direct or indirect links).

In the following sections, we will focus on the stability measure and discuss correlation between the stability of a formal concept and the relevance of its translation as an ontological concept.

3. Filtering of Relevant Concepts based on Stability

3.1. **Stability.** The idea of stability has been used to assess plausibility of hypotheses of different kinds. In this line of thought, [10] has introduced the realization of the idea of stability of hypotheses based on similarity of object descriptions, and has extended it to the formal concepts. Accordingly, in this paper, we will study the utility of this measure for the evaluation of the relevance of formal concepts by taking into account the two following points: (1) Richer structures; and (2) Ontological context. In our work, we use the definition of stability as follows :

Definition 1. Let *K=(O,A,I)* be a formal context and *(X,Y)* be a formal concept of *K*. The stability index, σ, of *(X,Y)* is defined as follows :

$$\sigma(X,Y) = \frac{|\{Z \subseteq X | Z' = X' = Y\}|}{2^{|X|}} \tag{1}$$

With : X, the set of objects (extent) and Y, the set of attributes (intent). The stability index of a concept indicates how much the concept intent depends on particular objects of the extent. In other terms, the stability index represents the probability for a concept to preserve its intent even if some objects of its extent disappear. The idea behind stability is that a stable intent is probably “real“ even if the description of some objects is “noisy“.

3.2. **Stability in an Ontological Context.** Below, we project stability in our ontological context and propose an interpretation of the result. In other terms, the idea is to determine the ontological qualities/characteristics that can be represented by stability.

In an ontology or a taxonomy, an abstraction is basically a grouping of sub-classes that share some properties. Then, the set of shared properties constitutes the description of the abstraction. If we focus on relevance as a specific component of the overall concept quality, then successfully approximating that quality by stability amounts to showing that the following assumption holds:

Assumption 1. Given a class *cl* with sub-classes $C = \{c_1, ..., c_n\}$ and described by a set of properties $P = \{p_1, ..., p_m\}$ then, the bigger the number of subsets $X \subseteq C$, such that the members of X share exactly the set of properties P, the higher the relevance of *cl*.

In the following sections, we present an experimental study that put this assumption to test. We present, first, our heuristic of filtering which underlies the experiments.

3.3. **Stability-Based Filtering.** Our filtering method has the following overall structure:

- **Inputs::** Relational Lattice Family generated by the RCA-based restructuring method.
- **Outputs::** Set of formal concepts (those deemed relevant).
- **Body::** Computation of the evaluation metrics.

The following is a simple heuristic based only on the stability measure. The principle of this heuristic is illustrated by the following rules. Given a formal concept FC,

- **Rule 1::** Extent cardinality = 1 (object concept); Stability is invariably 0.5. The concept is assumed **relevant**.
- **Rule 2::** Extent cardinality = 2; Stability is either 0.25 (two sub-concepts) or 0.5 (single sub-concept). In this case the stability value is unrelated to the relevance, so the case is deemed **inconclusive**.
- **Rule 3::** Extent cardinality $\geqslant 3$; Stability value in the unit interval. A threshold criterion is applied as follows:

Rule 3.1:: If value $\geqslant 0.5$ the concept (a stable one[1]) is deemed **relevant**, otherwise, it is **irrelevant**.

4. Implementation and Validation Study

Below we present the experimental study aimed at testing the **Assumption 1**.

4.1. **Software Environment.** Our approach was implemented within the Inukhuk platform [16] which is a service-oriented infrastructure based on RCA with a set of tools for *Ontological Engineering.* Inukhuk includes services for ontology construction, modularization, merge and restructuring. Inukhuk is coupled with Galicia platform providing RCA services, as well as other platforms and APIs (e.g. Jena, Wordnet, Wikipedia, Gate, Align, SimPak).

An initial work [15] attempted to validate the RCA-based restructuring framework. Experiments have been carried out on medium-size ontologies which confirmed the satisfactory performances of the tool in terms of reorganization and identification of new abstractions. These showed the limits of the "naive" lattice filtering algorithm that was initially implemented and thus underscored the need for more effective filtering strategies.

Our filtering tool, called *RLFDistiller*, receives as input an RLF and outputs a set of relevant formal concepts. To date, three metrics are implemented: stability and density to evaluate the structural relevance, and semantic similarity with object concepts to evaluate the semantic relevance.

4.2. **Experiments.** An appropriate validation of the approach should follow an outline like: (1) Selection of a set of poorly designed ontologies; (2) Application of our restructuring tool to each of these ontologies and generation of relational lattices; (3) Evaluation by human experts of formal concepts representing the new discovered abstractions; (4) Application of our filtering tool on relational lattices; and (5) Correlation between filtering results and experts judgements.

However, due to the difficulties in obtaining poorly structured yet plausible ontologies and the scarcity of domain experts eager to collaborate in such experiments, we proceeded as follows:

(1) **Selection of a set of good quality ontologies** (complete and without redundancies) which will be considered as *reference ontologies.*
(2) **Focused perturbations of selected ontologies**. In order to generate poorly designed ontologies we introduce *redundancies* and *incompleteness.* Incompleteness comes from removing non leaf classes

[1]Please notice that the threshold of 0.5 is the one used in the literature.

from the class hierarchy. In doing that, we nevertheless preserve the properties –datatype and object ones– and property restrictions of the class that disappears: Whenever necessary, these are transferred to the sub-classes[2]. In this way, some redundancy may be generated. The resulting ontologies are called *degraded ontologies*. It is noteworthy that due to the way FCA-based restructuring works, all classes in a degraded ontology will appear in the *restructured ontology*.

(3) **Construction of relational lattices**. The restructuring tool is applied to each degraded ontology and relational lattices are generated. Such a lattice includes at least the following three categories of formal concepts representing, respectively: (1) initial concepts that are kept in the degraded ontology; (2) "missing abstractions" (removed concepts) rediscovered by the tool; and (3) new abstractions discovered by the tool (no equivalent in the reference ontology). In general, such abstractions could be deemed either relevant or not, by an expert. However, we intentionally chose complete ontologies so that the only relevant abstractions discovered by the tool correspond to concepts we removed.

(4) **Application of *RLFDistiller* on relational lattices**. The tool outputs a set of relevant formal concepts to become the *restructured ontology*.

(5) **Confrontation of each *restructured ontology* with its *reference ontology***. In order to measure the accuracy and completeness of our approach, we use the *precision* and *recall* measures as defined in Information Retrieval [13]. They require, in turn, the calculation of the following four sets:
   - **True Positives (TP)**: Concepts from the restructured ontology that have an equivalent in the reference ontology.
   - **False Positives (FP)**: Concepts from the restructured ontology that have no equivalent in the reference ontology.
   - **True Negatives (TN)**: Formal concepts from the relational lattice deemed irrelevant by the tool that have no equivalent ontological concept in the reference ontology.
   - **False Negatives (FN)**: Formal concepts from the relational lattice deemed irrelevant by the tool that have an equivalent ontological concept in the reference ontology.

[2]Obviously, this way to perturb ontologies ensures that all removed abstractions will be discovered by the RCA-based tool.

TABLE 2. Statistics on the *reference ontologies*.

Ontology name	Classes #	Object prop. #	Datatype prop. #	Individuals #
Cms	6	5	10	0
Travel	34	6	4	14
People	60	14	1	21
Tourism	76	26	27	111

TABLE 3. Examples of perturbations.

Ontology	Perturbation
Travel_degraded	1. Remove the *RuralArea* class and move its property *id_RuralArea* to the subclasses. 2. Remove the *Activity* class and move its properties *hasActivity* and *isOfferedAt* to the subclasses.
People_degraded	1. Remove *Company* class and transfer its property *id_company* to the subclasses. 2. Remove *Publication* class and transfer its properties *id_publication* and *reads* to the subclasses.

We applied the above protocol to a number of small/medium-sized ontologies whose size-oriented metrics are provided in Table 2. Table 3, in turn, exemplifies typical focused perturbations.

Now, in the analysis of the results generated by our tool, we followed a three-fold question as stated below. The goal was to verify to which extent:

(1) Concepts from the degraded ontology have high stability values ($\geqslant 0.5$).
(2) Removed abstractions are represented by formal concepts with high stability values ($\geqslant 0.5$).
(3) Other discovered abstractions (no equivalent in the reference ontology) have low stability values (< 0.5).

4.3. **Experimental Results.** The outcome of our experimental study are summarized in Table 4 which provides the respective cardinalities for the above four sets and for each ontology. It also cites the values of the quality metrics. Below, we illustrate the four cases.

- **TP :** In the relational lattice of the Travel ontology:
  - Concepts $c_{\#5}$ and $c_{\#23}$ (see Fig. 5) which represent two abstractions in the degraded ontology (Destination and UrbanArea, respectively) were deemed relevant (stability values of 0.91 and 0.68, respectively).

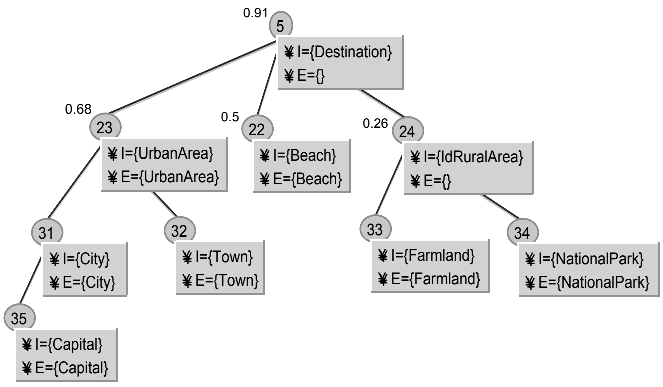

FIGURE 5. A part of concept lattice of Travel ontology.

- Concept $c_{\#43}$ (see Fig. 6) representing the removed abstraction Activity emerged by factoring out shared links to properties; due to its high stability (0.68), it was labeled relevant by the tool.

- **TN :** In the Travel ontology again (see Fig. 6) concepts $c_{\#45}$, $c_{\#42}$ and $c_{\#46}$ represent newly discovered abstractions. In the lattice, they are also super-concepts of $c_{\#43}$ (Activity). For instance, $c_{\#45}$ groups Destination and Activity, hence it corresponds to an overly general notion. All three concepts are deemed irrelevant due to low stability (0.46, 0.48 and 0.49, respectively).
- **FP :** Concept $c_{\#66}$ from the lattice of the People ontology (see Fig. 7) has a stability of 0.75 and is therefore labeled relevant by the tool. However, the abstraction that it represents is too general so it doesn't exist in the reference ontology (grouping subclasses of Adult with subclasses of Publication).
- **FN :** Concept $c_{\#8}$ with the People ontology (see Fig. 7) represents the removed abstraction Publication that was indeed rediscovered. Thus it is a legitimate concept to keep in the restructured ontology. However, it was filtered out by the tool since its stability of 0.46 is below the threshold.

4.4. **Discussion.** From the above results, the following partial conclusions could be drawn: First, there is clearly no perfect correlation between stability of a concept and its relevance. In a slightly more negative tone, stability could not even be used to order concepts w.r.t relevance: Higher stability does not necessarily mean higher relevance.

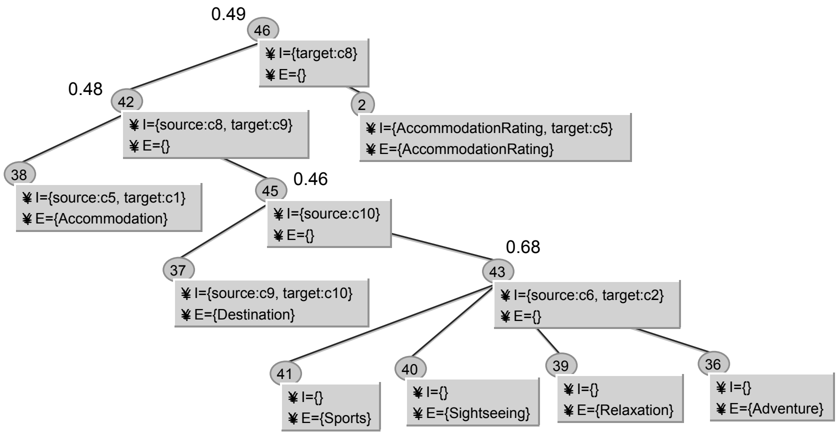

FIGURE 6. A part of concept lattice of Travel ontology.

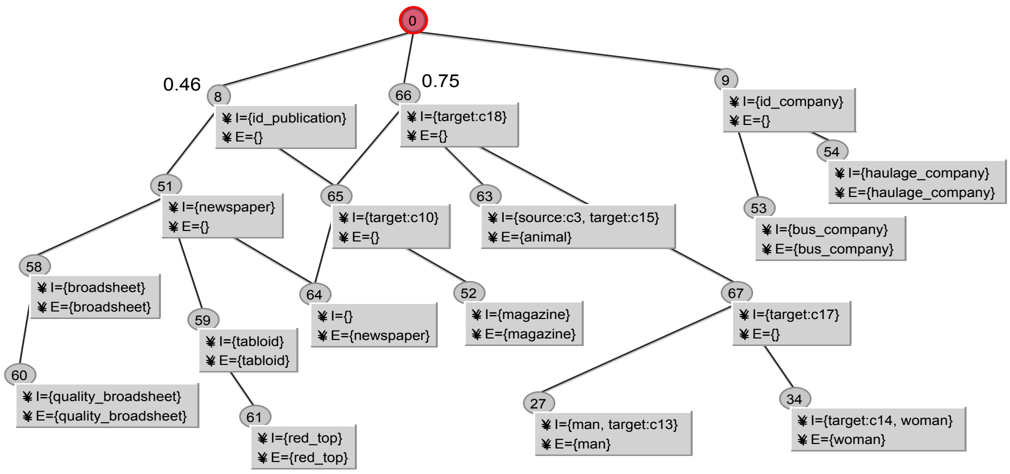

FIGURE 7. A part of concept lattice of People ontology.

On the positive side, 100% of the concepts in the degraded ontologies got high stability scores and therefore were deemed relevant by the tool. This speaks in favor of using stability as a component in the target filtering heuristic, possibly complemented by other measures.

TABLE 4. Final Results of experiments.

Ontology name	TP	FP	TN	FN	Recall	Precision	F-score
Cms	4	0	2	2	0.66	1	0.80
Travel	33	0	3	1	0.91	1	0.95
People	59	2	3	1	0.95	0.96	0.95
Tourism	75	9	21	1	0.78	0.89	0.83

Next, 37.5% of the rediscovered abstractions were labeled relevant[3]. While this rate might seem low, it is worth noting that another 50% got an *inconclusive* label. In theory, these could be retrieved by a different metric in the final filtering tool. Hence the real recognition rate for missing abstractions as to the current study is between 37.5% and 87.5%. However, in the assessment of our tool (see table 4), we took a conservative approach and we re-labeled all inconclusive cases as irrelevant.

Finally, spurious abstractions discovered by the RCA were filtered out by the tool with a 51.25% rate whereas another 24.39% were deemed inconclusive (again chances are they got stricken by another relevance metric). Thus, the overall pruning rate for irrelevant abstractions lays between 51.25% and 75.64%.

As a general conclusion, in the light of this first batch of experiments, concept stability seems to be a fair approximation of the relevance in an ontological context. Moreover, we tend to see the performances of our filtering tool as satisfactory and encouraging, in particular, w.r.t. to the improvements it brings to the complete ontology restructuring tool.

5. RELATED WORK

FCA has been successfully applied to problems arising with class hierarchy design, maintenance and refinement [4, 6]. Moreover, several studies have used FCA as part of a process of ontology engineering. In [12], the authors have introduced a FCA-based methodology for the design of formal ontologies for product families. Terms describing the components in a product family along with the required properties are captured in a lexicon set and put into context. A class hierarchy is then derived from the concept lattice and exported into OWL. [18] have explored ontology construction by merging two existing ontologies provided with a corpus of textual documents. NLP techniques are used to capture the relationships between documents and classes from an ontology and organize them into a dedicated context. The two contexts are then merged and a pruned concept lattice is constructed which is

[3]Recall that the tool will invariably discover all of them

further transformed into a merged ontology by a human expert. The pruned concept lattice is computed with the algorithm TITANIC. This algorithm computes the formal concepts via their key sets (or minimal generators). A key set is a minimal description of a formal concept. These key sets are used, firstly, to indicate if the generated formal concept gives rise to a new concept in the target ontology or not. A concept is new if and only if it has no key sets of cardinality one. Secondly, the key sets of cardinality two or more can be used as generic names for new concepts and they indicate the arity of new relations.

The notion of stability has been used to prune concept lattices, notably, in the fields of social network analysis for dealing with communities. The goal in [11] is to select potentially relevant concepts based on their stability degrees. The method has been applied to large datasets from other domains as well. In [14], the authors apply FCA as a representation framework for the knowledge about the structure of a given knowledge community (based on shared vocabulary and topics of interests). Stability has been used here to prune the lattice so that only a sub-hierarchy thereof could be kept, comprising the most interesting concepts.

6. Conclusion and Future Work

Our ultimate goal is to improve the structural and semantic quality of an ontology. To that end, we study a restructuring approach based on a relational extension of FCA. Here, we focus on a crucial stage of the restructuring process, i.e., the filtering of the concepts from the output lattices and tackle the question of assessing their relevance. Relevance being contextual and subjective, we are studying various metrics to approximate it.

In this paper, we examine the stability measure and attempt an evaluation of its usefulness for our goals. We thus carried out a row of experiments in which we took an existing ontology, purposefully degraded its quality – structure and completeness–, run our restructuring tools, applied stability-based filtering, and finally compared the resulting set of concept deemed relevant to the initial ontology. The results of the experiments seem to reveal the following picture: While the experimental hypothesis of a good correlation between stability and relevance is not universally valid, which is hardly a surprise. Yet if we restrict the evaluation to formal concepts whose extents have *at least three* formal objects (i.e., sufficiently general), the correlation greatly improves.

The next step of the experimental study would be to put the apparent threshold of 50% that seems to work relatively well to test: Could this value be the manifestation of a general phenomenon (e.g., related to the definition of stability) or is it a consequence of a bias in our choice of ontologies for the experiments. On the technical side, further relevance measures are currently under examination, in particular, ones that bring in some semantics either by exploiting the input ontology or by exploring external sources (upper ontologies, structured vocabularies, etc.). We believe that the ultimate filtering method should rely on a combination of such measures, hence at a more advanced stage the exact form of that combination should be investigated.

References

[1] H. Alani, C. Brewster: *Metrics for ranking ontologies*, In: Evaluating Ontologies for the Web Workshop (EON2006), 15th International World Wide Web Conference, 23-26 May 2006, pp. 11-17.

[2] M. Barbut, B. Monjardet: *Ordre et classification: algebre et combinatoire*. Vol- ume 2. Hachette Paris, 1970.

[3] J. Baumeister, D. Seipel: *Verification and refactoring of ontologies with rules*. In: Managing Knowledge in a World of Networks. Springer (2006) pp. 82-95.

[4] M. Dao, M. Huchard, M.R. Hacene, C. Roume, P. Valtchev: *Improving general- ization level in uml models iterative cross generalization in practice*. In: Conceptual Structures at Work. Springer (2004) pp. 346-360.

[5] B. Ganter, R. Wille: Formal Concept Analysis. Mathematical Foundations. Springer, Berlin-Heidelberg-New York, 1999.

[6] R. Godin, P. Valtchev: *Formal concept analysis-based class hierarchy design in object-oriented software development*. In: Formal Concept Analysis. Springer (2005) pp. 304-323.

[7] A. Gomez-Perez: *Ontological engineering: A state of the art*. Expert Update: Knowledge Based Systems and Applied Artificial Intelligence 2(3) (1999) pp. 33-43

[8] T.R. Gruber, et al.: *A translation approach to portable ontology specifications*. Knowledge acquisition 5(2) (1993) pp. 199-220.

[9] S.M. Henshaw, A. El-Masri, P.A. Sage: *Restructuring ontologies through knowl- edge discovery*. In: E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services, IEEE (2008) pp. 441-444.

[10] S.O. Kuznetsov: *On stability of a formal concept*. Annals of Mathematics and Artificial Intelligence 49(1-4) (2007) pp. 101-115.

[11] S. Kuznetsov, S. Obiedkov, C. Roth.: *Reducing the representation complexity of lattice-based taxonomies*. In: Conceptual Structures: Knowledge Architectures for Smart Applications. Springer (2007) pp. 241-254.

[12] J. Nanda, T.W. Simpson, S.R. Kumara, S.B. Shooter: *A methodology for prod- uct family ontology development using formal concept analysis and web ontology language*. Journal of computing and information science in engineering 6 (2006) pp. 103-113.

[13] C. Rijsbergen: *Information retrieval*. Butterworths (1975)

[14] C. Roth, S., Obiedkov, D., Kourie, D.: *Towards concise representation for taxonomies of epistemic communities*. In: Concept Lattices and their Applications. Springer (2008) pp. 240-255.

[15] M. Rouane-Hacene, S. Fennouh, R. Nkambou, P. Valtchev: *Refactoring of on- tologies: Improving the design of ontological models with concept analysis*. In: Tools with Arti cial Intelligence (ICTAI). Volume 2., IEEE (2010) pp. 167-172.

[16] M. Rouane-Hacene, P. Valtchev, R. Nkambou: *Supporting ontology design through large-scale fca-based ontology restructuring.* In: Conceptual Structures for Discovering Knowledge. Springer (2011) pp. 257-269.
[17] M. Rouane-Hacene, M. Huchard, A. Napoli, P. Valtchev: *Relational concept analysis: mining concept lattices from multi-relational data.* Annals of Mathematics and Arti cial Intelligence (2013) pp. 1-28
[18] G. Stumme, A. Maedche: *Fca-merge: Bottom-up merging of ontologies.* In: IJCAI. Volume 1. (2001) pp. 225-230.
[19] M.C. Suarez-Figueroa, A. Gomez-Perez: *First attempt towards a standard glossary of ontology engineering terminology.* In the 8th International Conference on Terminology and Knowledge Engineering (2008).
[20] O. Svab-Zamazal, V. Svatek, C. Meilicke, H. Stuckenschmidt: *Testing the impact of pattern-based ontology refactoring on ontology matching results.* In: The 7th International Semantic Web Conference, Citeseer (2008) pp. 240-271.

Department of Computer Science, UQAM, Montreal, Canada
E-mail address: fennouhs@gmail.com
E-mail address: nkambou@gmail.com
E-mail address: rouanehm@gmail.com
E-mail address: valtchev.petko@uqam.ca