Introducing metrics in the lattice to build ontology

My Thao TANG and Yannick TOUSSAINT

Loria-Campus Scientifique,
BP 239 - 54506 Vandoeuvre-les-Nancy Cedex, France
{My-Thao.Tang,Yannick.Toussaint}@loria.fr

Abstract. Formal Concept Analysis (FCA) ([1]) which is a formal conceptualisation method offers several advantages for building ontology. The hierarchical structure of the lattice resulting from FCA suits well the need for organizing classes in an ontology from general to most specific concepts. The lattice can be considered as the knowledge model from which, domain experts can select the concepts to build ontology. However, the task evaluation of the lattice can be impractical if domain experts have to evaluate all the concepts and it is not easy to handle if they want to change something in the lattice to make it more in accordance with their needs. In this paper, we conduct a study of using metrics in the lattice to build ontology. The metrics are introduced in the lattice to support domain experts in the tasks evaluation and refinement. In this work, firstly, we learn various metrics that are defined for ontology evaluation and lattice; secondly, we apply them to analyze the lattice, identify the set of metrics that can work for our goal and propose the way of using them to support efficiently evaluation and refinement of the lattice. The experiment results confirm that the use of the metrics is helpful to facilitate the lattice evaluation and refinement.

Keywords: Formal Concept Analysis, Building Ontology, Metrics

1 Introduction

Formal Concept Analysis (FCA) ([1]) is a formal conceptualisation method which has proved very efficient for building ontologies ([2–5]). It has often been highlighted for its ability at conceptualizing data in a *bottom-up* process and can exploit all necessary elements for the representation of knowledge: individuals, attributes and relationships between individuals. The hierarchical structure of the lattice resulting from FCA suits well the need for organizing classes in an ontology from general to most specific concepts. The lattice can be considered as the knowledge model of the domain, from which experts could select the concepts that fit their needs for building the final ontology. However, any formal methods have disadvantages when modeling a cognitive process that is not governed by clear and precise rules. Domain experts may be not satisfied with the knowledge model produced automatically. There are several reasons for experts to ask for changes in the lattice: (1) there may be noise in resources or in the information extraction processes and thus, some concepts result from this noise; (2) experts are not satisfied with the resulting knowledge model and wish it to be more in accordance with their needs, *i.e.* the application that will use the knowledge model. Researches ([6–9]) try to overcome this problem by proposing the lattice to domain experts for evaluation and refinement. Thus, experts evaluation and refinement are important tasks that are needed in a lattice-based ontology building process.

In this paper, we present an approach introducing metrics in the lattice to build ontology. Metrics are introduced in the lattice to help domain experts in evaluating and refining the lattice in a more efficient way. Evaluation and refinement of the lattice are practical and essential, but not easy tasks. The question is, which concept should we start evaluating? The task of evaluation becomes overwhelming for domain experts if they have to waste effort on checking many concepts that are not "important" for them. Experts may want to make some changes in the lattice to make it more in accordance with their needs, but they don't know if changing a concept can lead to delete some "important" concepts. Several possible strategies can be applied to execute a change, which strategy should be chosen? We address these problems by introducing metrics in the lattice in order to facilitate the evaluation and refinement of the lattice. For this purpose, we analyze the lattice based on a set of metrics to give experts the insight of the lattice. With the help of the metrics, experts can efficiently select the concepts to evaluate and recognize the concepts should be refined. Moreover, the metrics helps experts in handling the refinement process. Based on the metrics value, experts can know if a change leads to delete "important" concepts and be aware if a change strategy can affect a large part of the lattice so that they can efficiently making decisions on changes. The approach is to bring the flexibility of use to the lattice-based ontology building process, support for more efficient evaluation and refinement. The contributions of this work are firstly, we analyze the metrics for ontology evaluation and lattice and identify the set of metrics that can be used for supporting domain experts in the tasks evaluation and refinement; secondly, we propose the way of using the metrics for these tasks; finally, we apply the approach to a real-world dataset and show that using the metrics in the lattice can help for evaluation and refinement.

The rest of the paper is organized as follows. Section 2 gives some basics of FCA. Section 3 describes the set of proposed metrics for analyzing the lattice. Section 4 presents the methodology of using the metrics to support the tasks evaluation and refinement of the lattice. Using metrics for the lattice refinement is developing in section 5. Then, in section 6, the experiment and discussion follow. Finally, section 7 concludes the paper.

2 Formal Concept Analysis basis

Formal Concept Analysis (FCA) [1] is a mathematical formalism for building a *concept lattice* from a *formal context* $\mathcal{K} = (G, M, I)$ where G denotes a set of objects, M a set of attributes and I a binary relation defined on $G \times M$. $I \subseteq G \times M$ is a binary table which assigns a attribute to an object (see Table 1).

A *concept* is a pair (A, B) where $A \subseteq G$ is called the *extent*, $B \subseteq M$ is called the *intent* of the concept, and A is the maximal set of objects sharing the whole set of attributes in B (and vice versa). Concepts are computed on the basis of a *Galois connection* defined by two derivation operators denoted by $'$:

$$A' := \{m \in M \mid \forall g \in A, gIm\}$$
$$B' := \{g \in G \mid \forall m \in B, gIm\}$$

A concept (A, B) verifies a closure constraint so that $A' = B$ and $B' = A$.

Formal concepts are organized into a complete concept lattice denoted by $\mathcal{L}$ following a partial ordering, called subsumption, $(\sqsubseteq)$ between a concept and a superconcept and defined as follows: $(A_1, B_1) \sqsubseteq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2$ (*or* $B_2 \subseteq B_1$). (A_1, B_1) is called *subconcept* of (A_2, B_2), and (A_2, B_2) is called *superconcept* of (A_1, B_1). The supremum of all the concepts in the lattice is called the *top* $(\top)$, and the infimum is called the *bottom* $(\bot)$.

Table 1 illustrates a formal context, $\mathcal{K}$, describing a set of objects about diseases and their attributes. The corresponding lattice, $\mathcal{L}$, is given in Fig. 1.

Object	OCCURS_IN_woman	ISA_rare_disease	COEXISTS_WITH_old_age	CAUSES_ischemia
fibromuscular_dysplasia	x	x		x
breast_disease	x			
hypertensive_disease	x		x	
bone_disease			x	x

Table 1: Binary context $\mathcal{K}$.

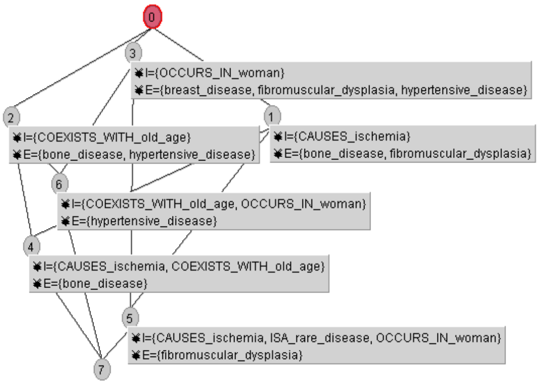

Fig. 1: The lattice $\mathcal{L}$.

3 The proposed metrics

With the aim of supporting domain experts in the tasks evaluation and refinement of the lattice, we would like to assist them in choosing the concepts for the evaluation process, identifying the concepts that may need to be refined, the concepts are potentially important and making decisions when they want to change something in the lattice. To that end, we have looked at the literate about the principles of designing ontological models ([10–12]) and the state of the art of the metrics that are defined for analyzing the quality of ontology and lattice. We collect the metrics proposed to access different features of a class in ontology (*class-level metrics*) and the metrics proposed to analyze concepts in the lattice, investigate which metrics can work for our purpose. The proposed metrics that are formalized within FCA are following.

1 - *Number Of Objects (NOO)*

Definition 1. For a given concept C in the lattice $\mathcal{L}$, NOO measures the number of objects that belong to this concept.

$NOO(C \in \mathcal{L}) = |\texttt{Extent(C)}|$.

This metric helps in identifying which concepts in the lattice that objects are in focus. The greater NOO value is, the more number of objects focuses on the concept. The NOO metric is the same as the ontology metric *class importance* that is introduced by [13]. In the lattice in Fig. 1, NOO value for the concept C_2 is 2.

2 - *Number Of Attributes (NOA)*

Definition 2. For a given concept C in the lattice $\mathcal{L}$, NOA measures the number of attributes of this concept.

$NOA(C \in \mathcal{L}) = |\texttt{Intent(C)}|$.

This metric shows how detailed the concept is described in term of attributes. The greater NOA value is, the more detailed the concept is described. In the lattice in Fig. 1,NOA value for the concept C_2 is 1.

3 - *Number of children (NOC)*

Definition 3. For a given concept C in the lattice $\mathcal{L}$, NOC measures the number of the subconcepts of C in the lattice.

$NOC(C \in \mathcal{L}) = |\{C_1 \in \mathcal{L} \mid C_1 \sqsubseteq C\}|$.

This metric shows the amount of subconcepts that depend on this concept. Those subconcepts may be affected when this concept is changed. A concept has several subconcepts is a sensitive concept and experts should pay a great attention for its refinement. The greater NOC value is, the more part of the lattice may be affected. The NOC metric is the same as the ontology metric *number of children* that is introduced by [14]. In the lattice in Fig. 1, NOC value for the concept C_2 is 2.

4 - *Number Of Siblings (NOS)*

Definition 4. For a given concept C in the lattice $\mathcal{L}$, NOS measures the number of the concepts that have the same superconcept with C in the lattice.

$NOS(C \in \mathcal{L}) = |\{C_1 \in \mathcal{L} \mid \exists C_2 \in \mathcal{L} : C \sqsubseteq C_2, C_1 \sqsubseteq C_2\}|$.

NOS metric shows the amount of concepts that depend on the same superconcept with this concept. Changes in the concept affect its superconcept and may affect the siblings. The greater NOS value is, the more siblings may be affected when this concept is changed. Similar to NOC, if a concept has no sibling or too many siblings, checking its superconcepts may be necessary. In the lattice in Fig. 1, NOS value for the concept C_2 is 2.

5 - *Number Of Parents (NOP)*

Definition 5. For a given concept C in the lattice $\mathcal{L}$, NOP measures the number of the superconcepts of C in the lattice.

$NOP(C \in \mathcal{L}) = |\{C_1 \in \mathcal{L} \mid C \sqsubseteq C_1\}|$.

The NOP metric shows the amount of superconcepts that affect this concept. If any of those superconcepts is changed, this concept may be affected. The greater NOP value is, the more concepts affect this concept and therefore, the maintenance of this concept is more complex. In the lattice in Fig. 1, NOP value for the concept C_2 is 1.

6 - *Depth of Inheritance Tree (DIT)*

Definition 6. For a given concept C in the lattice $\mathcal{L}$, DIT measures the length of the longest path from C to the *top* of the lattice.

The DIT metric indicates the abstraction level of the concept in the lattice. The greater DIT value is, the deeper the abstraction level of the concept is and the more concepts are inherited from this concept. Changes in any of the inherited concepts may affect this concept. The greater DIT value indicates the evaluation and maintenance of this concept are more complex. DIT metric is the same as the ontology metric *depth of inheritance* that is introduced by [14]. In the lattice in Fig. 1, DIT value for the concept C_2 is 1.

7 - *Centrality Measure (CM)*

Definition 7. For a given concept C in the lattice $\mathcal{L}$, CM how far concept C is near the middle level of the lattice.

$CM(C \in \mathcal{L}) = 1 - \left| \frac{\frac{l}{2} - DIT(C)}{\frac{l}{2}} \right|$

where

$l = max(\{\forall C_j \in \mathcal{L} \rightarrow DIT(C_j)\})$

According to ([11]), middle level concepts tend to be more detailed and prototypical of their abstractions than concepts at higher or lower abstraction levels in the hierarchy, and therefore, more "important". The greater CM value is, the nearer the middle level of the lattice the concept is. This metric is the same as the ontology metric *centrality measure* proposed by [15]. In the lattice in Fig. 1, the maximum DIT value is 3, we get the CM values for the concept C_2 and for the concept C_6 as follows.

$CM(C_2) = 1 - \left| \frac{\frac{3}{2} - DIT(C_2)}{\frac{3}{2}} \right| = 1 - \left| \frac{\frac{3}{2} - 1}{\frac{3}{2}} \right| = 0.667$

8 - *Density Measure (DM)*

Definition 8. For a given concept C in the lattice $\mathcal{L}$, DM how much concept C is rich conceptual neighborhood (children, siblings, parent).

Let $D = \{D[1], D[2], D[3]\} = \{NOC(C), NOS(C), NOP(C)\}$, w_i be a weight factor.

$DM(C \in \mathcal{L}) = \sum_{i=1}^{3} w_i \frac{D[i]}{max(\{\forall C_j \in \mathcal{L} \rightarrow D[i](C_j)\})}$

The greater DM value is, the richer conceptual neighborhood the concept is. This metric is adapted from the ontology metric *density measure* that is introduced by ([16]) and *localDensity* that is introduced by ([17]) that consider the neighborhood of a concept. Values of the metrics are normalized to be in the range [0-1] by dividing by the maximum metric value for all the concepts.

9 - *Intensional Stability (IS)*

Intensional stability which is a metric in the lattice is firstly described by [18] and later in [19, 20] as a way to measure the probability of preserving its intent after dropping some objects. According to the [19, 20], concepts with high intensional stabilities are likely typical of realistic categories; concepts relying on noisy objects are more likely to be intensionally unstable.

$IS(C \in \mathcal{L}) = \frac{|\{A \subseteq Extent(C) | A' = Intent(C)\}|}{2^{|Extent(C)|}}$

10 - *Extensional Stability (EI)*

Similar to intensional stability, extensional stability [19, 20] measures the probability of preserving its extent after dropping some attributes. According to [19, 20], concepts relying on noisy attributes will be extensionally unstable.

$ES(C \in \mathcal{L}) = \frac{|\{A \subseteq Intent(C) | A' = Extent(C)\}|}{2^{|Intent(C)|}}$

We describe how to use the metrics to help domain experts in evaluation and refinement following the principles of designing ontological models in the next section.

4 Methodology

In this section, we present the way of using the metrics to support the tasks evaluation and refinement of the lattice. First, we propose the procedures based on the metrics for evaluation. Then, we show how we apply the metrics to help experts in the evaluation process.

4.1 Evaluation procedures

Based on the view of the approaches for defining concepts for an ontology ([12]), we propose three procedures for the task evaluation of the lattice: *top-down*, *bottom-up* and *middle-out*. The approach for defining concepts for an ontology depends on the view of the experts about the domain. Top-down approaches start with the most general concepts in the domain first and then specialize them. Bottom-up approaches start with the most specific concepts, the leaves of the hierarchy and then organize them into more general concepts. Middle-out approaches start from the fundamental concepts, then generalize and specialize them. None of these approaches is better than any of the others. For experts that have top-down view of the domain, they would prefer the most general concepts to start evaluating, we propose to them a top-down evaluation procedure. Bottom-up evaluation procedure is proposed to start evaluating from the most specific concepts for experts that have bottom-up view of the domain. Middle-out evaluation procedure is proposed to start evaluating from the concepts that are nearest the middle level of the lattice for experts that would prefer to start from the fundamental concepts.

The concepts are proposed to experts for evaluating according to the metric values. With the use of the metrics in the lattice, each concept in the lattice is attached to a set of metric values. For the top-down evaluation procedure, the concepts that are nearest the *top* of the lattice (the most general level) are proposed to experts for evaluating first; then, their subconcepts follow. Experts can choose an objective metric with respect to their preference.The objective metric can be NOO, NOA, NOC, NOS, NOP, DM, IS or ES. These concepts are proposed to experts in the descending order of the objective metric metric values. For example, experts prefer the concepts that have many objects and choose the objective metric NOO for evaluation. Then, the concept with the

Algorithm 1 Top-down evaluation procedure

procedure TOP-DOWNEVALUATION(**In:** $\mathcal{L}$ a lattice, m the objective metric)
  $\{m \in \{NOO, NOA, NOC, NOS, NOP, DM, IS, ES\}$
  Local: $S, S1$ first-in-last-out stacks
  Initialize the status of all the concepts to *pending*;
  Put the subconcepts of the *top* to S in the ascending order of the metric m values;
  repeat
    $C \leftarrow S.pop()$;
    Propose C for evaluating and set $C.status$ to *checked*;
    Put the subconcepts of C that are not *checked* to $S1$ in the ascending order of the metric m values;
    repeat
      $C1 \leftarrow S1.pop()$;
      Propose $C1$ for evaluating and set $C1.status$ to *checked*;
      Put the subconcepts of $C1$ that are not *checked* and not in $S1$ to $S1$ in the ascending order of the metric m values;
    until $S1$ is empty
  until S is empty
end procedure

largest NOO value, which gathers the largest number of objects, is proposed for starting the evaluation procedure. The detailed description of the top-down evaluation procedure is given in Algorithm 1. In the evaluation procedure, when the system proposes a concept to experts for evaluating, if the experts don't care about the concept, they can skip it or they can ask for some changes to make it more accordance with their needs.. If experts decide to make another iteration, the lattice will be updated to take into account the changes. If experts decide to stop the evaluation, the procedure will end. A concept that is already visited is marked as *checked* to avoid proposing it for evaluation several times.

For the bottom-up evaluation procedure, the concepts that are nearest the *bottom* of the lattice (the most specific level) are proposed first; then, their superconcepts follow. Experts can choose an objective metric. The concepts are proposed to experts in the descending order of the objective metric values. For the middle-out evaluation procedure, the concepts which are nearest the middle level of the lattice are proposed first; then, their superconcepts and subconcepts follow. Experts also can choose an objective metric. The concepts are proposed to experts in the descending order of the CM metric values, which measures how far a concept is near the middle level of the lattice, then the objective metric value.

4.2 Support for refinement

When a concept is proposed to experts for evaluating, according to the metric values of the concept, the experts can get some suggestions for the refinement. The suggestions for refining the concept regard the set of objects, the set of attributes and its conceptual structure, we have:

– If a concept has a low intensional stability (IS value), it may rely on noisy objects ([19, 20]), then experts should check the extent to remove the noisy objects.

- If a concept has a low extensional stability (ES value), it may rely noisy attributes ([19, 20]), then experts should check the intent to remove the noisy attributes.
- If a concept has only one child ($NOC = 1$), there may be a modelling problem and therefore, checking this concept may be necessary.
- If there are more than a dozen subconcepts for a given concept ($NOC \geq 12$), then experts should consider adding additional intermediate concepts ([10]) or removing some subconcepts.
- If a concept has a high NOS value (too many siblings), checking its superconcepts may be necessary.

The suggestions for refining the structure of a concept are based on the guidelines for designing a class hierarchy suggested by [10]. They are:

If a class has only one direct subclass there may be a modeling problem or the ontology is not complete.

If there are more than a dozen subclasses for a given class then additional intermediate categories may be necessary.

All the siblings in the hierarchy (except for the ones at the root) must be at the same level of generality.

5 Using metrics for lattice refinement

Here, we present the way of using metrics for lattice refinement. To that end, we propose a metric to help experts make decisions when changing the lattice.

One way for refining the lattice is by changing the data. Several reasons may motivate experts for refining the lattice. However, a unique lattice is produced for a set of input data. Some concepts in the lattice are not "ontological" but no concept can be removed from the lattice. If experts do not want to accept those as concepts of the ontology, they cannot be actually removed from the lattice. The idea is, experts interact on the lattice, but changes are performed on data. Objects and attributes that are used to build the lattice can be stored in a set of annotations. Once experts make changes for refining the lattice, the data (annotations) is updated to ensure that, the lattice gives a new conceptualisation, closer to experts' needs.

To assist experts in specifying their wishes for changing the lattice, in the previous work [7], we define a set of operations for making changes in the lattice (ex: *remove concept* c). Several possible *strategies* can be applied to execute a change. When experts ask for a change, a list of possible strategies are suggested to them. Once the expert makes a choice of the strategies, the data is updated and the lattice is modified accordingly. For example, experts evaluate the lattice given in Fig. 1, concept c_6 is not good for them and they ask for removing this concept. The strategies for removing concept are suggested for refining the set of attributes in the intent of the concept by removing or adding attributes by the help of the superconcepts and subconcept. Table 2 shows an example of the strategies for removing concept c_6 from the initial lattice given in Fig. 1. Here,

ID	Strategies
S1	Removing attribute `OCCURS_IN_woman` from concept c_6
S2	Removing attribute `COEXISTS_WITH_old_age` from concept c_6
S3	Adding attributes `ISA_rare_disease` and `CAUSES_ischemia` to concept c_6

Table 2: Strategies for removing concept c_6.

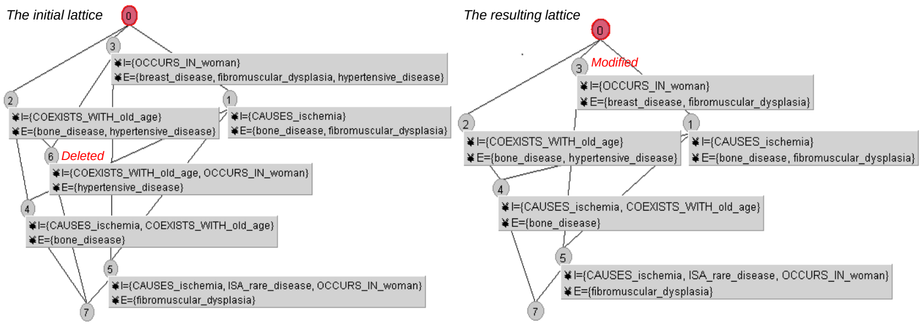

Fig. 2: The initial lattice and the resulting lattice after removing attribute `OCCURS_IN_woman` from concept c_6 (strategy $S1$).

three strategies are suggested for refining the set of attributes in the intent of concept c_6 by removing or adding attributes according to the superconcepts,c_2 and c_3, the subconcept c_7.

The question is, which strategy should be chosen for a change? A change strategy can lead to modify, delete some other concepts or create some new concepts in the lattice. Can the metrics support experts in making decisions on changes and protecting *important* concepts from changes?

The new lattice $\mathcal{L}^*$ after a change is obtained from the existing lattice $\mathcal{L}$ by taking, deleting, modifying some concepts and creating new concepts. These induced changes are called *impact* of a change strategy. We distinguish four possible types of concepts, *old* concepts, *deleted* concepts, *modified* concepts and *new* concepts, as follows. Let C be a concept in $\mathcal{L}^*$:

- C is an *old* concept if there exists a concept in $\mathcal{L}$ that has the same extent and intent to C,
- C is a *modified* concept if there exists a concept in $\mathcal{L}$ that has the same intent to $\mathtt{Intent(C)}$ but the extent is different from $\mathtt{Extent(C)}$,
- C is a *new* concept if $\mathtt{Intent(C)}$ doesn't exist in $\mathcal{L}$,
- C in $\mathcal{L}$ is a *deleted* concept if $\mathtt{Intent(C)}$ doesn't exist in $\mathcal{L}^*$.

Fig. 2 shows an example of removing concept c_6 according to the strategy $S1$, removing attribute `OCCURS_IN_woman` from concept c_6. In this example, concept c_6 in the initial lattice, is *deleted*; concept c_3 is *modified*, lost the object `hypertensive_disease` from its extent; the other concepts are *old*.

To support experts in making decisions on changes, we propose an *Impact Index (II)* of a change strategy for indicating how much a change strategy affect the lattice. The impact index of a change strategy is computed by summing

ID	*Impact index*	*modified*	*new*	*deleted*
S1	0.5	c_3		c_6
S2	0.75			c_2,c_6
S3	0.75	c_1,c_4,c_5		c_2,c_6

Table 3: Strategies for removing concept c_6 associated with the impact indexes and the details of the concept that are modified, new or deleted.

the intensional stability of deleted concepts. The idea is to measure how many stable intent concepts can be lost from a change strategy. The larger value of the impact index is, the more number of stable intent concepts of the lattice will be lost. Impact index of a change strategy S is formalized as below.

$$II(S) = \sum_{\{C \in \mathcal{L} | C \texttt{ a deleted concept}\}} IS(C)$$

Once experts ask for a change, a list of possible strategies with their impact index is suggested to them. Table 3 shows an example of the strategies for removing concept c_6 associated with the impact indexes. The strategy with smallest impact index value, which will lead to loose the smallest number of stable intent concepts in the lattice, will be suggested first. In the example, strategy $S1$ with the minimum impact index, which lead to loose only one concept, c_6, is consider the best strategy for this change. Experts can choose a strategy for the change based on the impact index. By using the metrics in the lattice, experts can archive better decisions on making changes.

6 Experiment and discussion

In this section, we present the results of applying the metrics for evaluation and refinement of the lattice on a real-world dataset. Further, we discuss the results of using the metrics.

6.1 Setup

A dataset about `Fibromuscular dysplasia of arteries` on medical domain was conducted by extracting abstracts from PubMed[1]. Table 4 shows the information of the dataset and the lattice built from this dataset.

Statistics	Dataset and the lattice
Number of abstracts	284
Number of objects	530
Number of attributes	573
Number of concepts	567

Table 4: Statistics of the dataset and the lattice built from this dataset

The preprocessing step, we used SemRep ([21]) for extracting objects and their information from the selected abstracts. SemRep identifies entities and

[1] http://www.ncbi.nlm.nih.gov/pubmed

SE|00000000||tx|1|text|Fibromuscular dysplasia of the renal arteries.

- SE|00000000||tx|1|entity|C0016052|Fibromuscular Dysplasia|dsyn|||Fibromuscular dysplasia||||1000|1|23
- SE|00000000||tx|1|entity|C0035065|Structure of renal artery|bpoc|||renal arteries ||||1000|32|45
- SE|00000000||tx|1|relation|1|1|C0035065|Structure of renal artery|bpoc|bpoc|||renal arteries||||1000|32|45|PREP|LOCATION_OF||25|26|1|1|C0016052|Fibromuscular Dysplasia |dsyn|dsyn|||Fibromuscular dysplasia||||1000|1|23

Fig. 3: An example of entities and relations extracted by SemRep

Statistics	*NOO*	*NOA*	*NOC*	*NOS*	*NOP*	*DIT*	*CM*	*DM*	*IS*	*ES*
Maximum	106	82	29	213	25	6	1	0.68	0.999	1
Third quartile	3	3	2	213	2	3	0.667	0.358	0.5	0.5
Median	2	2	1	13	1	2	0.667	0.08	0.5	0.5
Minimum	1	1	1	0	1	1	0	0.025	0	0
Average	2.876	2.883	1.811	86.34	1.736	2.127	0.578	0.179	0.474	0.487

Table 5: Statistics of the metrics data

relations that can be found in the UMLS (Unied Medical Language System) Metathesaurus and UMLS Semantic Network. Some additional information is provided in the same line such as the preferential UMLS term and positions in the texts. Fig. 3 shows an example of entities and relations extracted by SemRep. In this example, *Fibromuscular Dysplasia* is extracted as an entity associated with the concept `C0016052` of UMLS; the preferential UMLS term is `Fibromuscular Dysplasia`. *renal arteries* is extracted as an entity associated with the concept `C0035065`; the preferential UMLS term is `Structure of renal artery`. `Structure of renal artery` has relation `LOCATION_OF` with `Fibromuscular Dysplasia` (the last line of the figure).

To encode in FCA a relation $\mathtt{r}(\mathtt{a},\mathtt{b})$ between two objects `a` and `b`, the relation `r` could be considered as an attribute of `a` (which is agent of the relation), and the inverse relation $\mathtt{r}^{-1}$ could be considered as an attribute of `b` (which is the object of the relation) ([22]). We adopted in our experiment a dissymmetric and more fine-grained description attached to the agent of the relation: the attribute $\mathtt{r} - \mathtt{b}$ is given to the object `a`. For example, `Structure of renal artery` was considered as an object with attribute `LOCATION_OF_Fibromuscular Dysplasia`.

6.2 Experiment results and discussion

We applied the metrics to evaluate and refine the resulting lattice above. In our experiment, the balance weights were set to 1/3 for computing metric *DM* since the importances of the number of children, siblings and parents were considered equal. Table 5 shows the statistics of the metrics data. In the statistics, we don't count the two especial concepts of the lattice, the *top* and the *bottom* because they are not "ontological".

Here, we present the experiment results and the observation of the suggestions and the change actions taken by domain experts to see how well the metrics work. Table 6 shows the ten iterations with the actions that were taken by domain experts in the top-down evaluation process with the objective metric NOO. When experts evaluate a concept, the metric values help them in suggesting refinement and making decision on changes in the refinement process. For example, in iteration 1, concept $C1$ (`intent={OCCURS_IN_patient}`) with the largest number of objects was proposed to experts for evaluating first. The metric IS with a very low value ($IS \sim 0$) helps experts in recognizing that the concept may contain noisy objects. Some noisy objects were removed from the concept and some objects were refined by adding attributes. The metric NOC with a very high value ($NOC = 29$) suggests experts that they should check the subconcepts to add some intermediate concepts or remove some subconcepts. When checking the subconcepts, the metric ES with a low value helps experts in finding noisy attributes. In this stage, they removed the noisy subconcepts, merged the subconcepts in which intents carried same meaning, removed noisy attributes and merged attributes in the subconcepts. One of them was subconcept $C19$ with $ES = 0.25$. Experts found this concept with noisy attributes (`intent={OCCURS_IN_patient,OCCURS_IN_journalist}`) and removed it. The impact index help experts in making decisions on changes. Table 7 shows the list of obtained concepts after the ten iterations of the evaluation procedure.

From the experiment results, we observed that the metrics adapted from ontology metrics are helpful in evaluating and refining the lattice but the ways they work on the lattice are quite different from on ontology because the lattice structure is different from the ontology structure. According to the authors [13], NOO metric shows the important of a concept because the concepts that have a lot of objects in the ontology are important. However, because the lattice has a lot of *joins* and *meets*, the joins often contain a lot of objects but not so meaningful for domain experts. For example, in the experiment (Table 6) , in iteration 1, the concept $C1$ with `intent=OCCURS_IN_patient`got the largest number of objects in the lattice but it was not so meaningful. The metrics NOC, NOS and NOP, can be used as hints but does not help much in refining the conceptual structure in the lattice. For example, in iteration 2, experts checked concept $C4$, NOC metric with value 1 tells the experts should check the subconcept if there is problem in modelling. In this iteration, the subconcept is a meet of concepts in the lattice, not a problem.

The lattice metrics, IS and ES, help in finding noisy objects and noisy attributes, but high stability does not mean the concept is meaningful for domain experts. For example, in iteration 6, concept $C2$ with a very low IS (`intent=ISA_symptom`) was meaningful for them.

From the observation, the lattice metrics should be considered in a combination with the ontology metrics in the evaluation and refinement process, they should not used as the criteria for eliminating concepts in the lattice. In the experiment, concept $C1$ (interation 1) with a very low IS and a high NOO gave interesting concepts after refinement. If we eliminate it because of low IS

Iter.	Concept	*NOO*	*NOA*	*NOC*	*NOS*	*NOP*	*DIT*	*CM*	*DM*	*IS*	*ES*	Change action
1	C1	106	1	29	213	1	1	0.333	0.68	~ 0	0.5	- Remove objects - Add attributes to objects - Remove subconcepts - Merge subconcepts - Merge attributes of subconcepts
	C19	5	2	3	31	2	2	0.667	0.11	0.657	0.25	
2	C4	40	2	1	23	2	2	0.667	0.084	0.002	0.25	- Remove objects
3	C5	1	5	1	1	2	5	0.333	0.051	0.5	0.437	- Remove attributes
4	C6	14	2	2	25	2	2	0.667	0.101	0.999	0.25	- Remove subconcept
5	C9	1	77	1	11	18	5	0.333	0.363	0.5	0	- Remove attributes - Merge attributes
6	C2	42	1	2	209	1	1	0.333	0.38	~ 0	0.5	- Remove objects
7	C11	19	1	4	209	1	1	0.333	0.405	0.967	0.5	- Remove objects - Add attributes - Add attributes - to subconcepts
8	C7	2	3	2	9	2	3	1	0.084	0.25	0.25	- Remove objects
9	C8	2	3	1	3	2	3	1	0.062	0.5	0.375	- Remove objects
10	C10	1	4	1	3	2	3	1	0.062	0.5	0.625	- Remove objects

Table 6: Iterations of the top-down evaluation procedure.

Concept	Intent
C1	`OCCURS_IN_patient`
C2	`ISA_symptom`
C3	`ISA_symptom, ISA_pain`
C4	`ISA_symptom,OCCURS_IN_patient`
C5	`ISA_symptom,OCCURS_IN_patient,COEXISTS_WITH_fibromuscular_dysplasia`
C6	`ISA_disease, OCCURS_IN_patient`
C7	`ISA_disease, OCCURS_IN_patient, OCCURS_IN_woman`
C8	`ISA_disease, OCCURS_IN_patient, CAUSES_hypertensive_disease`
C9	`ISA_disease,OCCURS_IN_patient,OCCURS_IN_woman_child,CAUSES_hypertensive_disease...`
C10	`ISA_disease,OCCURS_IN_patient,OCCURS_IN_female_child,COEXISTS_WITH_hamartoma`

Table 7: List of evaluated concepts after 10 iterations.

value, we will loose a lot of important objects and therefore, loosing interesting concepts further.

The impact index of change strategy is helpful for experts in making decisions on changes, but because it relies on intensional stability, it doesn't suggest the best strategy when the concepts with low intensional stabilities are important for experts. If the experts are interested in some concepts, the metric should take into account the the domain experts' opinions to help them in protecting the important concepts from changes. For example, in iteration 7, when refining concept $C11$ with `intent={ISA_disease}`, 5 strategies were suggested; experts chose the strategy adding attribute `intent=OCCURS_IN_patient` to concept $C11$ ($II = 0.969$) which was not the best suggested strategy because concept with `intent={OCCURS_IN_patient,ISA_disease}` is more meaningful for them.

6.3 Revising the algorithms and impact index

We have revised the evaluation procedures to take into account the domain experts' opinions. Algorithm 2 is revised from the top-down evaluation procedure in Algorithm 1. At the beginning of the evaluation procedure, all the concepts are initialized with the quality *normal* (line 3 in the Algorithm 2). When the

Algorithm 2 Top-down evaluation procedure

1: **procedure** RevisedTop-downEvaluation(**In:** $\mathcal{L}$ a lattice, m the objective metric)
$\{m \in \{NOO, NOA, NOC, NOS, NOP, DM, IS, ES\}$
2: **Local:** $S, S1$ first-in-last-out stacks
3: Initialize the status of all the concepts to *pending* and the quality to *normal*;
4: Put the subconcepts of the *top* to S in the ascending order of the metric m values;
5: **repeat**
6: $C \leftarrow S.pop()$;
7: Propose C for evaluating and set $C.status$ to *checked*;
8: **if** Experts consider C important **then**
9: Set $C.quality$ to *important*;
10: **end if**
11: Put the subconcepts of C that are not *checked* to $S1$ in the ascending order of the metric m values;
12: **repeat**
13: $C1 \leftarrow S1.pop()$;
14: Propose $C1$ for evaluating and set $C1.status$ to *checked*;
15: **if** Experts consider $C1$ important **then**
16: Set $C1.quality$ to *important*;
17: **end if**
18: Put the subconcepts of $C1$ that are not *checked* and not in $S1$ to $S1$ in the ascending order of the metric m values;
19: **until** $S1$ is empty
20: **until** S is empty
21: **end procedure**

system proposes a concept to experts for evaluating, if the experts don't care about the concept, they can skip it; if they are interested in the concept, they can mark its quality as *important* (lines 8-10 and 15-17).

The *Impact Index (II)* is revised to take into account the effect of important concept by domain experts. A concept which is important is considered the most stable intent concept ($IS = 1$). A change strategy that leads to delete important concepts will lead to loose more stable intent concepts than the others. By this way, experts can avoid the strategies that lead to delete the important concepts for them.

The revised impact index a change strategy S is formalized as below.

$$II(S) = |\{C \in \mathcal{L} \mid C \texttt{ a deleted and important concept}\}| + \sum_{\{C \in \mathcal{L} \mid C \texttt{ a deleted and not important concept}\}} IS(C).$$

The test of the new versions of the evaluation procedures and the impact index gave the better results with respect to experts' requirements.

7 Conclusion

In this paper, we have presented an approach using metrics in the lattice to build ontology. The metrics are introduced in the lattice to support domain experts in the tasks evaluation and refinement. With the information from the metric values of the concepts, experts can efficiently select the concepts for evaluation and identify the concepts that may needs to be refined. We have presented in this work a set of metrics and showed how them can be used for the lattice evaluation and refinement. Moreover, we propose a metric using intensional stability of concepts to estimate how much a change strategy will impact the lattice for better handling the refinement. Revising evaluation procedures and impact index of a change strategy are proposed to take into account the opinions of domain experts in the evaluation and refinement process. The experiment results confirm that the use of the metrics is helpful to facilitate the lattice evaluation and refinement. As future work, we plan to enhance the metrics to build lattice-based ontology. Taking into account new concepts and the problem of changing stability of concepts the impact index of a change strategy should be considered.

References

1. Ganter, B., Wille, R.: Formal Concept Analysis, Mathematical Foundations. Springer (1999)
2. Poelmans, J., Ignatov, D., Kuznetsov, S., Dedene, G.: Formal concept analysis in knowledge processing: A survey on applications. Expert Syst. Appl. **40**(16) (2013) 6538–6560
3. Bendaoud, R., Napoli, A., Toussaint, Y.: Formal concept analysis: A unified framework for building and refining ontologies. In Gangemi, A., Euzenat, J., eds.: EKAW. Volume 5268 of Lecture Notes in Computer Science., Springer (2008) 156–171
4. Cimiano, P., Hotho, A., Staab, S.: Learning concept hierarchies from text corpora using formal concept analysis. Volume Volume 24., USA, AI Access Foundation (August 2005) 305–339
5. Obitko, M., Snásel, V., Smid, J.: Ontology design with formal concept analysis. In: Proceedings of the CLA 2004 International Workshop on Concept Lattices and their Applications, Ostrava, Czech Republic (September 2004)
6. Fennouh, S., Nkambou, R., Valtchev, P., Rouane-Hacene, M.: Stability-based filtering for ontology restructuring. In: Proceedings of the 12th International Conference on Formal Concept Analysis - ICFCA 2014. (2014)
7. Tang, M., Toussaint, Y.: A collaborative approach for fca-based knowledge extraction. In: The Tenth International Conference on Concept Lattices and Their Applications - CLA'13, La Rochelle, France (September 2013)
8. Bendaoud, R., Toussaint, Y., Napoli, A.: Pactole: A methodology and a system for semi-automatically enriching an ontology from a collection of texts. In: Proceedings of the 16th international conference on Conceptual Structures: Knowledge Visualization and Reasoning. ICCS '08, Berlin, Heidelberg, Springer-Verlag (2008) 203–216
9. Stumme, G., Maedche, A.: Fca-merge: Bottom-up merging of ontologies. In: 17th International Joint Conferences on Artificial Intelligence (IJCAI'01), San Francisco, CA, Morgan Kaufmann Publishers, Inc. (2001) 225–234
10. Noy, N.F., McGuinness, D.L.: Ontology development 101: A guide to creating your first ontology. Technical report, Stanford Knowledge System laboratory, Stanford University, USA (2001)
11. Rosch, E.: Principles of Categorization. Cognition and Categorization. Lawrence Erlbaum Publishers, Hillside, NJ (1978)
12. Uschold, M., Gruninger, M.: Ontologies: Principles, methods and applications. Knowledge Engineering Review **11** (1996) 93–136
13. Tartir, S., Arpinar, I., Shethh, A.: Ontological evaluation and validation. In Poli, P., Healy, M., Kameas, A., eds.: Theory and Applications of Ontology: Computer Applications. Springer Netherlands (2010) 115–130

14. Zhang, H., Li, Y.F., Tan, H.: Measuring design complexity of semantic web ontologies. The Journal of Systems and Software **83**(5) (May 2010) 803–814
15. Alani, H., Brewster, C.: Ontology ranking based on the analysis of concept structures. In: Proceedings of the 3rd International Conference on Knowledge Capture. K-CAP '05, New York, NY, USA, ACM (2005) 51–58
16. Alani, H., Brewster, C., Shadbolt, N.: Ranking ontologies with aktiverank. In Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L., eds.: The Semantic Web - ISWC 2006. Volume 4273 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2006) 1–15
17. Peroni, S., Motta, E., d'Aquin, M.: Identifying key concepts in an ontology, through the integration of cognitive principles with statistical and topological measures. In Domingue, J., Anutariya, C., eds.: The Semantic Web. Volume 5367 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2008) 242–256
18. Kuznetsov, S.: On stability of a formal concept. Annals of Mathematics and Artificial Intelligence **49**(1-4) (2007) 101–115
19. Roth, C., Obiedkov, S., Kourie, D.: Towards concise representation for taxonomies of epistemic communities. In: Proceedings of the 4th International Conference on Concept Lattices and Their Applications. CLA'06, Berlin, Heidelberg, Springer-Verlag (2008) 240–255
20. Klimushkin, M., Obiedkov, S., Roth, C.: Approaches to the selection of relevant concepts in the case of noisy data. In Kwuida, L., Sertkaya, B., eds.: Formal Concept Analysis. Volume 5986 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2010) 255–266
21. Rindflesch, T.C., Fiszman, M.: The interaction of domain knowledge and linguistic structure in natural language processing: Interpreting hypernymic propositions in biomedical text. Journal of Biomedical Informatics **36** (2003) 462–477
22. Chekol, M., Alam, M., Napoli, A.: A study on the correspondence between fca and eli ontologies. In: The Tenth International Conference on Concept Lattices and Their Applications - CLA'13, La Rochelle, France (September 2013)