**ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA**
**CONSIGLIO NAZIONALE DELLE RICERCHE**

**A. Formica**

# ONTOLOGY-BASED CONCEPT SIMILARITY IN FORMAL CONCEPT ANALYSIS

**R. 633 Maggio 2005**

**Anna Formica** – Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti" del CNR, Viale Manzoni 30 - 00185 Roma, Italy. Email : anna.formica@iasi.cnr.it

**This paper appears in Information Sciences 176(18), pp.2624-2641, 2006**.

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica, CNR

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: iasi@iasi.rm.cnr.it
URL: http://www.iasi.rm.cnr.it

## Abstract

Both domain *ontologies* and *Formal Concept Analysis* (FCA) aim at modeling concepts, although with different purposes. In the literature, a promising research area concerns the role of FCA in ontology engineering, in particular, in supporting the critical task of reusing independently developed domain ontologies. With this regard, the possibility of evaluating *concept similarity* is acquiring an increasing relevance, since it allows the identification of different concepts that are semantically close. In this paper, an ontology-based method for assessing similarity between FCA concepts is proposed. Such a method is intended to support the ontology engineer in difficult activities that are becoming fundamental in the development of the Semantic Web, such us ontology *merging* and ontology *mapping* and, in particular, it can be used in parallel to existing semi-automatic tools relying on FCA.

**Keywords**: *Formal Concept Analysis, Semantic Web, domain ontologies, similarity reasoning.*

## 1. Introduction

Both domain *ontologies* and *Formal Concept Analysis* (FCA) aim at modeling concepts, although with different purposes. The purpose of a domain ontology is to model a "shared understanding of the domain of interest" [51], where "shared" means that an ontology captures consensual knowledge, i.e., accepted by a panel of experts in the given domain. On the other hand, the purpose of FCA [25, 53] is to support the user in analyzing and structuring a domain of interest. Given a domain, a concept in FCA is a pair of sets: a set of objects, which are the instances of the concept in that domain, and a set of attributes, which are the descriptors of the concept. Note that the extensional and intensional aspects are both important in FCA, whereas ontologies emphasize on the intensional component only. In fact, ontologies can be defined without objects, whereas FCA always relies on some set of objects.

In the literature different directions are being explored about possible interactions among FCA and Conceptual Modelling [42], Artificial Intelligence (in particular Description Logics) [1], Object-Oriented databases [54], and software engineering [50]. FCA techniques are also revealing interesting in supporting the development of the Semantic Web [9] and, in particular, ontology engineering [30, 2]: they can be used to extract from a given domain a conceptual hierarchy which may serve as a basis for the manual or semi-automatic development of an ontology [16]. Furthermore, due to the presence in the Web of large and specialized ontologies, FCA can also be used for the critical task of reusing and combining independently developed domain ontologies [48]. With this regard, the possibility of assessing similarity between concepts, also referred to as *similarity reasoning* [23], is becoming fundamental in the development of the Semantic Web, in particular to perform ontology *mapping*, *integration*, or *alignment* (these are only a few keywords found in the literature [32]). In fact, these are difficult activities that, in general, require human interaction and, therefore, are time-consuming and error-prone.

In this paper, a method for computing similarity of FCA concepts is proposed. Such a method is intended to support the ontology engineer in reusing existing ontologies and, in particular, it can be used in parallel to existing semi-automatic tools relying on FCA, as for instance [46, 48, 49]. The starting point of the approach is the selection of an existing domain ontology to identify *similar* concept names, with the related *similarity degrees*, as established by a panel of experts in the given domain [38]. On the basis of the similarity degrees, the method allows the evaluation of FCA concept similarity by taking into account both the intensional (the set of attributes) and the extensional (the set of objects) components of FCA concepts.

Note that, in general, from a theoretical point of view, ontology concepts are identified with FCA concepts [2]. However, in many applications, the canonical match is between ontology concepts and FCA attributes, that is the approach followed in this paper. Therefore, FCA attributes can be seen as concepts, in the sense that for building concepts, other concepts are needed that play the role of attributes [16].

The paper is organized as follows. In the next section the notion of a *Concept Lattice* is briefly recalled, which is used in FCA to organize and structure concepts. In Section 3, domain ontologies are recalled, with particular attention to the notion of a *similarity graph*. Successively, in Section 4, the method for evaluating concept similarity in Concept Lattices is presented, followed by the Related Work Section. Finally, Section 6 concludes the paper with some hints about future work.

## 2. Formal Concept Analysis

FCA provides a conceptual framework for structuring, analyzing and visualizing data, in order to make them more understandable [53]. It is based on *lattice theory* [10], a well established mathematical discipline that has been applied within many different realms, like Psychology, Sociology, Medicine, Linguistics, and Computer Science. In FCA, application domains are organized and structured according to *Concept Lattices*, also referred to as *Galois Graphs*.

Concept Lattices are constructed by first identifying the relevant objects of a given application domain, together with their relevant features. In this perspective, a concept is not an abstraction but, on the basis of the observation of the reality, it is a clustering of objects and related common attributes.

### 2.1. Concept Lattices

In FCA a concept is defined within a *context*. A context is a triple $(O,A,R)$, where $O$ and $A$ are two sets of elements called *objects* and *attributes*, respectively, and $R$ is a binary relation between $O$ and $A$. In particular, if $oRa$, for $o \in O$ and $a \in A$, then we say that "the object $o$ has the attribute $a$" or "the attribute $a$ applies to the object $o$".

Given two sets $E$, $I$, such that $E \subseteq O$ and $I \subseteq A$, consider the *dual* sets $E'$ and $I'$, i.e., the sets defined by the attributes applying to all the objects belonging to $E$ and the objects having all the attributes belonging to $I$, respectively, that is:

$$E' = \{a \in A \mid oRa \ \forall \ o \in E\}$$
$$I' = \{o \in O \mid oRa \ \forall \ a \in I\}$$

Then, a *concept* of the context $(O,A,R)$ is a pair $(E,I)$ such that $E \subseteq O$, $I \subseteq A$ and the following conditions hold:

$$E' = I,\ I' = E.$$

The sets $E$ and $I$, representing the concept extensional and intensional components respectively, are referred to as the *extent* and the *intent* of the concept, respectively. Therefore, a concept is a pair of sets where the former consists of precisely those objects which have all attributes from the latter and, conversely, the latter consists of precisely those attributes that apply to all objects from the former.

From a philosophical point of view, according to the theory of [53, 25], "a concept is a unit of thoughts consisting of two parts, the *extension* and the *intension*. The extension covers all objects (or entities) belonging to the concept, while the intension comprises all attributes (or properties) valid for all those objects." Therefore, in general, the sets of objects and attributes are expected to be disjoint. In other words, in FCA "objects and attributes are two types of items that relate to each other in an application, and the use of the terms *object* and *attribute* is indicative. They could also be referred to as, for instance, *documents* and *terms*. The main idea is that enlarging a set of, for instance, terms, will reduce the set of documents containing all these terms, whereas a smaller set of terms will match a larger set of documents" [42].

For instance, consider a context called *European Cities* where:

$$O = \{\text{Athens, Courmayeur, Innsbruck, London, Paris, Reykjavik, Rome}\},$$

$A$ = {Archeological_Site, Beach, Capital, Euro, River, Skiing_Area}

and $R$ is specified by the Table 1, where *Arc*, *Bea*, *Cap*, *Eur*, *Riv* and *Ski* stand for *Archeological_Site*, *Beach*, *Capital*, *Euro*, *River*, and *Skiing Area*, respectively.

| | Arc | Bea | Cap | Eur | Riv | Ski |
|---|---|---|---|---|---|---|
| Athens (A) | x | x | x | x | | |
| Courmayeur (C) | | | | x | | x |
| Innsbruck (I) | | | | x | x | x |
| London (L) | | | x | | x | |
| Paris (P) | | | x | x | x | |
| Reykjavik (Re) | | | x | | | x |
| Rome (Ro) | x | x | x | x | x | |

Table 1: The *European Cities* context

In this context, seven objects are present, each corresponding to a European city, and six attributes. A concept of this context is, for instance, the pair:

((Athens,Paris,Rome), (Capital,Euro))

that is, in short form:

((A,P,Ro), (Cap,Eur))

In fact, all of *Athens*, *Paris*, and *Rome* have both the *Capital*, and *Euro* attributes, and viceversa *Capital*, and *Euro* together apply to no other object than *Athens*, *Paris*, and *Rome*. Intuitively, it is possible to say that concepts correspond to maximal rectangles of crosses in the context, after appropriate permutations of rows and columns.

Note that, given a context ($O$,$A$,$R$) and two concepts ($E_1$,$I_1$) and ($E_2$,$I_2$), the following conditions hold:

if $E_1 \subseteq E_2$ then $E_2' \subseteq E_1'$, for $E_1$, $E_2 \subseteq O$
if $I_1 \subseteq I_2$ then $I_2' \subseteq I_1'$, for $I_1$, $I_2 \subseteq A$,

that is, duality implies the opposite set inclusion for both objects and attributes. Therefore, as already mentioned, by adding attributes to a concept (i.e., by identifying additional discriminating attributes), the cardinality of its extent decreases, and viceversa, by adding objects to a concept the cardinality of its intent decreases.

Given two concepts ($E_1$,$I_1$), ($E_2$,$I_2$) of a context ($O$,$A$,$R$), it is possible to establish an *inheritance relation* ($\leq$) between them according to the following condition:

($E_1$,$I_1$) $\leq$ ($E_2$,$I_2$) iff $E_1 \subseteq E_2$ (iff $I_2 \subseteq I_1$).

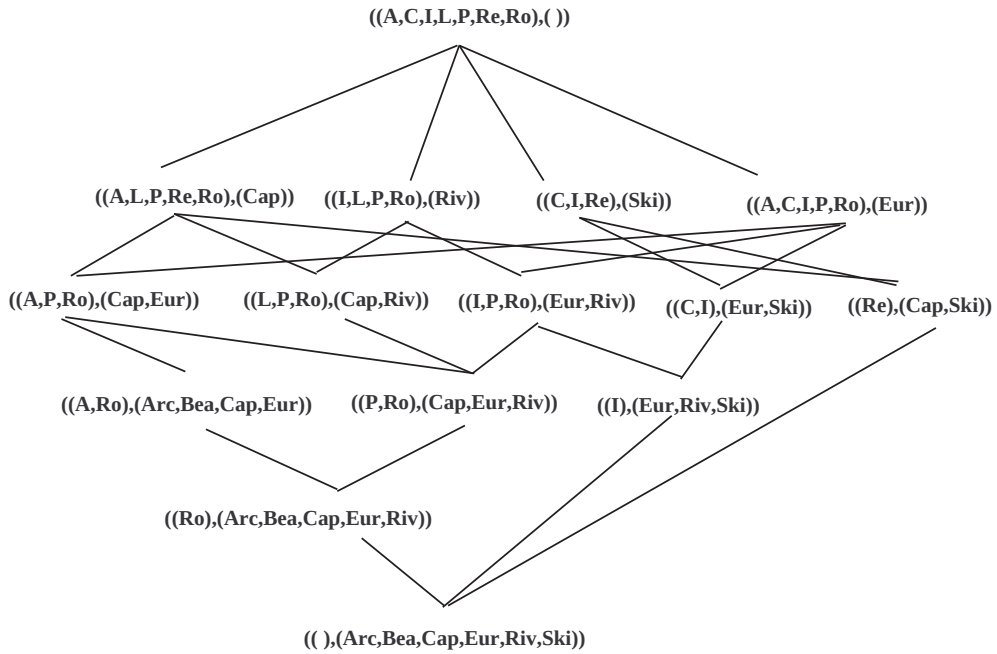


Figure 1: Concept Lattice of the *European Cities* context

In particular, $(E_1,I_1)$ is called *subconcept* of $(E_2,I_2)$ and $(E_2,I_2)$ is called *superconcept* of $(E_1,I_1)$. Note that inheritance relation has been extensively addressed in Conceptual Modelling [12], with particular attention to Object-Oriented databases (see, for instance, [4, 22]).

Given a context $(O,A,R)$, consider the set of all concepts of this context, indicated as $\mathcal{L}(O,A,R)$. Then:

$$(\mathcal{L}(O,A,R),\leq)$$

is a complete lattice called *Concept Lattice* (also referred to as *Galois Graph*), i.e., for each subset of concepts, the greatest lower bound and the lowest upper bound exist [53]. (Note that for lattices over sets with finite cardinality, the notions of complete lattice and lattice coincide [10].)

In a Concept Lattice, nodes are labeled with the concepts of the context, and arcs are established among the nodes whose associated concepts are in $\leq$ relation. By the definition, the $\leq$ relation is a partial order relation that expresses a double inclusion among node components. In particular, given a node, say $n$: (i) the extent of $n$ is contained in the extent of each of the ancestors of $n$, and (ii) the intent of $n$ contains the intent of each of the ancestors of $n$.

The Concept Lattice has also two special nodes, the maximum and minimum nodes (labeled with $\top$ and $\bot$, respectively). The maximum and the minimum group all the objects and the attributes of the context, respectively.

For instance, the Concept Lattice that can be constructed from the context of Table 1 is shown in Figure 1.

Note that the representation of a Concept Lattice can be optimized. In fact, by assuming

that, for each node, attributes are inherited from the top ($\top$) and objects are inherited from the bottom ($\bot$), Concept Lattices can be optimized by defining the related *Inheritance Graph*, whose nodes contain only the additional elements, objects and attributes, with respect to the descendants and ancestors, respectively. However, the optimized representation of a Concept Lattice will not be addressed in this paper and, for reader's convenience, all the objects and attributes of a concept will be explicitly given.

## 3. Domain Ontology

We have seen in the Introduction that a domain ontology is a "shared understanding of the domain of interest" [51], where "shared" means that the ontology definitions are accepted by a panel of experts in the given domain. The following definition is also worthwhile [20]: an ontology is a "formal, explicit specification of a shared conceptualization", where a "conceptualization" is an abstract model of some phenomenon of the world which identifies the relevant concepts (or entities) and relationships among the concepts of that phenomenon. Again, "shared" means that an ontology captures consensual knowledge, whereas "formal" refers to the fact that an ontology should be machine-understandable. Therefore, a domain ontology contains a set of interrelated concepts, each associated with a formal definition providing an unambiguous meaning of the concept in the given domain.

As already mentioned in the Introduction, *similarity reasoning* is acquiring interest in the development of the Semantic Web, in particular for ontology mapping and ontology integration [32]. In this perspective, below we recall the notion of a domain ontology as defined in [23], the paper that has inspired this proposal. Note that the following definition has been simplified by focusing on the aspects that are relevant to this work, i.e., the *similarity* relation.

**Definition 3.1. [Domain ontology]** A *domain ontology* (*ontology*, for short) $\mathcal{O}$ is specified by a set of entity[1] names $E_{\mathcal{O}}$, and a set of semantic relations, such us *generalization* (*ISA*), *partOf*, *relatedness*, *similarity* etc.. In particular, the *similarity* relation is a ternary relation:

$$similarity(c_i, c_j, as(c_i, c_j))$$

where $c_i$,$c_j$ are entity names, and $as(c_i, c_j)$ is a decimal number in the interval [0.0 ... 1.0] standing for the *axiomatic similarity degree* (*axiomatic similarity*, for short) of the entities $c_i$,$c_j$ according to the ontology $\mathcal{O}$. Such a degree is established by means of a *Consensus System* by a panel of experts in the given domain [38]. □

The above definition allows us to present the notion of a *similarity graph*, that is the basis for the similarity evaluation method proposed in this paper. Such a notion, that has been originally introduced in [23], has been revisited and modified in order to deal with FCA concepts. As already mentioned in the Introduction, in this paper the approach generally adopted in many applications for combining FCA and ontologies has been followed [2]: given a domain ontology and a context ($O$,$A$,$R$), the similarity graph is constructed by integrating ontology entities and FCA attributes.

**Definition 3.2. [Similarity graph]** Given a domain ontology $\mathcal{O}$, and a context ($O$,$A$,$R$), consider the set $E_{\mathcal{O}}$ of entity names of the ontology and assume that $\mathcal{C} = E_{\mathcal{O}} \cup A$.

[1] In ontology definitions the term *concept* is generally used in place of that of *entity*. However, in this paper we prefer to use *entity* to avoid confusion with the notion of a *concept* in a Concept Lattice.

Then, let $\Gamma_{(\mathcal{O},A)}$ be the set of triples $\langle c_i, c_j, as(c_i, c_j)\rangle$, where $c_i$,$c_j \in \mathcal{C}$ and $as(c_i, c_j)$ is a decimal number, such that the following conditions hold:

- for each pair $c_i$,$c_j \in \mathcal{C}$ such that $similarity(c_i, c_j, as(c_i, c_j))$ holds in the ontology $\mathcal{O}$, then $\langle c_i, c_j, as(c_i, c_j)\rangle \in \Gamma_{(\mathcal{O},A)}$;
- for any $c \in \mathcal{C}$:
  $\langle c, c, as(c, c)\rangle \in \Gamma_{(\mathcal{O},A)}$, where $as(c, c) = 1.0$
- for any pair $c_i, c_j \in \mathcal{C}$ for which the axiomatic similarity is not defined in the ontology $\mathcal{O}$:
  $\langle c_i, c_j, as(c_i, c_j)\rangle \in \Gamma_{(\mathcal{O},A)}$, where $as(c_i, c_j) = 0.0$
- for any pair $c_i, c_j \in \mathcal{C}$ :
  $as(c_i, c_j) = as(c_j, c_i)$ (symmetry of the axiomatic similarity) and
  $\langle c_i, c_j, as(c_i, c_j)\rangle \equiv \langle c_j, c_i, as(c_j, c_i)\rangle$.

The set $\Gamma_{(\mathcal{O},A)}$ is referred to as the *similarity graph* of the ontology $\mathcal{O}$ and the context $(O,A,R)$, and for any $\langle c_i, c_j, as(c_i, c_j)\rangle \in \Gamma_{(\mathcal{O},A)}$, $as(c_i, c_j)$ will be referred to as the *axiomatic similarity* of $c_i$,$c_j$ according to the similarity graph $\Gamma_{(\mathcal{O},A)}$. □

For instance, consider the context $(O,A,R)$ of the *European Cities* and suppose to have a simple domain ontology $\mathcal{O}$ where the following *similarity* relation holds:

$similarity(City, Capital, 0.8)$
$similarity(River, Stream, 0.9)$
$similarity(Beach, Seaside, 0.9)$

The similarity graph $\Gamma_{(\mathcal{O},A)}$ of this ontology and the *European Cities* context is given by the following set of triples:

$\langle City, Capital, 0.8\rangle$
$\langle River, Stream, 0.9\rangle$
$\langle Beach, Seaside, 0.9\rangle$
$\langle City, City, 1.0\rangle$
$\langle Archeological\_Site, Archeological\_Site, 1.0\rangle$
...
$\langle City, Archeological\_Site, 0.0\rangle$
$\langle City, Beach, 0.0\rangle$
...

where only some of the triples with similarity degrees equal to 1.0 and 0.0 have been given for short.

## 4. Concept Similarity

In this section the notion of *similarity* (*Sim*) between FCA concepts is introduced. With regard to the comparison of the intensional components of the concepts, the approach has been inspired by the method for evaluating concept similarity defined within the Enterprise Ontology Management Tool *SymOntos* [23] (successively, *SymOntoX* [36]). Essentially, it is based on the

*maximum weighted matching* problem in bipartite graphs, that can be solved in polynomial time [24]. Informally, it is illustrated as follows.

Consider a domain ontology $\mathcal{O}$ and two concepts ($E_1$,$I_1$) and ($E_2$,$I_2$) not necessarily belonging to the same context, and suppose, for instance, they belong to the contexts ($O_1$,$A_1$,$R_1$) and ($O_2$,$A_2$,$R_2$), respectively. Let $\Gamma_{(\mathcal{O},A_1 \cup A_2)}$ be the similarity graph of the ontology $\mathcal{O}$ and the given contexts. Then:

- consider the cartesian product $I_1 \times I_2$;
- let a *candidate set of pairs* be a subset of $I_1 \times I_2$ such that there are no two pairs in the set sharing an element. For instance, assume that $I_1$ and $I_2$ represent a set of boys and a set of girls, respectively, a candidate set of pairs defines a possible set of marriages (when polygamy is not allowed) [24]. Then, consider the set of all candidate sets of pairs, that is indicated as $\mathcal{P}(I_1, I_2)$;
- for each candidate set of $\mathcal{P}(I_1, I_2)$, consider the sum of the axiomatic similarity degrees of the pairs of attributes according to the similarity graph $\Gamma_{(\mathcal{O},A_1 \cup A_2)}$;
- the candidate set having the maximal among all the computed sums is chosen.

Then, as shown by Definition 4.2, the similarity of two concepts is essentially given by the weighted average between the cardinality of the intersection of the extents of the concepts and the maximal sum above.

**Definition 4.1. [The set of candidate sets of pairs]** Consider two concepts ($E_1$,$I_1$) and ($E_2$,$I_2$) of one or more contexts. Let $n$,$m$ be the cardinalities of the sets $I_1$, $I_2$, respectively, i.e. $n = |I_1|$, $m = |I_2|$, and suppose that $n \leq m$. The set $\mathcal{P}(I_1, I_2)$ of the *candidate sets of pairs* is defined by all possible sets of $n$ pairs of attributes defined as follows:

$$\mathcal{P}(I_1, I_2) = \{\ \{\langle a_1, b_1\rangle \ \ldots\ \langle a_n, b_n\rangle\} \mid a_h \in I_1,\ b_h \in I_2,\ \forall\ h = 1 \ldots n,\ \text{and } a_h \neq a_k,\ b_h \neq b_l,\ \forall\ k,l \neq h\}.$$

□

**Definition 4.2. [Concept similarity (Sim)]** Consider a domain ontology $\mathcal{O}$, and one or more contexts ($O_i$,$A_i$,$R_i$), $i = 1 \ldots k$.

The *concept similarity* (*Sim*) of two concepts ($E_1$,$I_1$) and ($E_2$,$I_2$) of the same (or different) context(s) is defined as follows:

$$Sim((E_1, I_1), (E_2, I_2)) = \frac{|\ (E_1 \cap E_2)\ |}{r} * w + \left[ \frac{1}{m} \max_{P \in \mathcal{P}(I_1, I_2)} \left( \sum_{\langle a,b \rangle \in P} as(a, b) \right) \right] * (1 - w)$$

where $\mathcal{P}$ and $m$ are defined as in the previous definition, $as(a, b)$ is the axiomatic similarity degree of $a$,$b$ according to the similarity graph $\Gamma_{(\mathcal{O},\cup_i A_i)}$, and $r$ is the greatest between the cardinalities of the sets $E_1$ and $E_2$. Finally $w$ is a weight such that $0 \leq w \leq 1$, that can be established by the user to enrich the flexibility of the method. □

Note that *Sim* is always a value between zero and one and, for any pair of concepts $(E_1, I_1)$, $(E_2, I_2)$, $Sim((E_1, I_1), (E_2, I_2)) = Sim((E_2, I_2), (E_1, I_1))$.

For instance, consider our running example, and two concepts of the European Cities context. In particular, consider the similarity graph of Section 3, and assume that $w = \frac{1}{2}$. Let us start

by evaluating the similarity of two sibling concepts of the Concept Lattice of Figure 1, namely $((L, P, Ro), (Cap, Riv))$, and $((A, P, Ro), (Cap, Eur))$. In this case, according to the similarity graph we have:

$\langle Capital, Capital, 1.0\rangle$ and $\langle River, Euro, 0.0\rangle$

therefore, since $r = 3$, and $m = 2$:

$$Sim[((L, P, Ro), (Cap, Riv)), ((A, P, Ro), (Cap, Eur))] = \\ \frac{2}{3} * \frac{1}{2} + \frac{1}{2} * (1.0 + 0.0) * (1 - \frac{1}{2}) = 0.58$$

Similarity increases if we consider a concept and one of its direct descendent (child) in the Concept Lattice. In fact, consider again the concept $((L, P, Ro), (Cap, Riv))$, and let us evaluate the similarity with the child $((P, Ro), (Cap, Eur, Riv))$. The following holds:

$$Sim[((L, P, Ro), (Cap, Riv)), ((P, Ro), (Cap, Eur, Riv))] = \\ \frac{2}{3} * \frac{1}{2} + \frac{1}{3} * (1.0 + 1.0) * (1 - \frac{1}{2}) = 0.66$$

Of course, similarity decreases in the case of concepts that are not directly related. For instance, consider one of the previous concepts, namely $((P, Ro), (Cap, Eur, Riv))$, and its ancestor $((A, L, P, Re, Ro), (Cap))$. Then:

$$Sim[((P, Ro), (Cap, Eur, Riv)), ((A, L, P, Re, Ro), (Cap))] = \\ \frac{2}{5} * \frac{1}{2} + \frac{1}{3} * 1.0 * (1 - \frac{1}{2}) = 0.36$$

Consider now the following concept belonging to a different context, say $World\ Cities$:

$$((Ro, Rio), (Sea, Cit, Str, Air))$$

where $Ro$, $Rio$, $Sea$, $Cit$, $Str$ and $Air$ stand for *Rome*, *Rio de Janeiro*, *Seaside*, *City*, *Stream*, and *Airport*, respectively. Consider the similarity graph of Section 3, suitable extended with the triples related to the attributes of the $World\ Cities$ context. Then, the similarity of this concept with, for instance, the concepts $((I, P, Ro), (Eur, Riv))$ and $((A, Ro), (Arc, Bea, Cap, Eur))$ of the Concept Lattice of Figure 1 are respectively:

$$Sim[((Ro, Rio), (Sea, Cit, Str, Air)), ((I, P, Ro), (Eur, Riv))] = \\ \frac{1}{3} * \frac{1}{2} + \frac{1}{4} * (0.9 + 0.0) * (1 - \frac{1}{2}) = 0.28$$

since $\langle Stream, River, 0.9\rangle$ holds, and:

$$Sim[((Ro, Rio), (Sea, Cit, Str, Air)), ((A, Ro), (Arc, Bea, Cap, Eur))] = \\ \frac{1}{2} * \frac{1}{2} + \frac{1}{4} * (0.9 + 0.8 + 0.0 + 0.0) * (1 - \frac{1}{2}) = 0.46$$

since $\langle Beach, Seaside, 0.9\rangle$ and $\langle City, Capital, 0.8\rangle$ hold. Note that the compared concepts have, in both the cases, one object only in common ($Ro$), whereas the number of pairs of attributes with non-null similarity degree is one in the first case ($Sream$ and $River$), and two in the second case ($Beach$, $Seaside$, and $City$, $Capital$). Analogously, let us compare the concept

$((Ro, Rio), (Sea, Cit, Str, Air))$ with the top (containing all the objects of the context and no attributes) and the bottom (containing all the attributes of the context and no objects) of the Concept Lattice of Figure 1. We have respectively:

$$Sim[((Ro, Rio), (Sea, Cit, Str, Air)), ((A, C, I, L, P, Re, Ro), ())] = \frac{1}{7} * \frac{1}{2} + \frac{1}{4} * 0 * (1 - \frac{1}{2}) = 0.07.$$

$$Sim[((Ro, Rio), (Sea, Cit, Str, Air)), ((), (Arc, Bea, Cap, Eur, Riv, Ski))] = 0 * \frac{1}{2} + \frac{1}{6} * (0.9 + 0.8 + 0.9 + 0.0) * (1 - \frac{1}{2}) = 0.22$$

Therefore, we can see that in evaluating the distance of concepts of Concept Lattices the role played by "similar attributes" is more important than the presence of common objects. Indeed, this is what we expect in evaluating concept similarity with the aim of performing, for instance, ontology integration or ontology mapping since, with respect to Concept Lattices, ontologies emphasize the intensional component, i.e., the descriptors of the concepts, rather than data.

## 5. Related Work

Similarity reasoning has been tackled in different fields of Computer Science, although a large majority of results have not been conceived for the Semantic Web, but rather for schema and data integration [40, 17]. In particular, the problem of integrating independently developed schemas (ontologies) into one single schema has been extensively investigated in the literature since the 1980s [3]. It consists in identifying the interschemas relationships that allow the reconciliation of the structure and terminology of the original schemas [15]. Currently, the more general problem of schema matching [44] is becoming fundamental in many applications, such as web-oriented data integration, electronic commerce, component-based development, etc.. In particular, with the development of the Semantic Web and the growing use of ontologies, the problem of overlapping knowledge in a common domain is becoming critical, and we are assisting to a big variety of terminology in this research field, as for instance, ontology *alignment*, *mapping*, *merging*, *translation*, *articulation*, that are just a few terms [32, 39].

Regarding ontology alignment, that consists in finding the corresponding entities in different ontologies, various frameworks and techniques have been proposed in the literature [21, 14, 34]. They focus on different kinds of ontology heterogeneity, in particular, syntactic, structural, and semantic heterogeneity. A comprehensive overview about ontology integration and ontology mapping can be found in [31, 52]. However, solving the problem of overlapping knowledge still requires human interaction, and is typically performed manually. It is, therefore, a time-consuming, error-prone and expensive activity.

In [33, 41], the importance of dealing with semantically heterogeneous data by using ontologies has been emphasized. In particular, according to [33], the methods and tools supporting ontology integration and maintenance can be divided according to two families, one based on Galois Lattices and the other one on Description Logics (DL). Regarding the methods based on DL, subsumption reasoning is generally used in order to compute relations among different information sources [14]. For instance, an ontology merging method has been proposed in [28], which is based on similarity relations among concepts represented according to DL. However, due to the focus of this paper on Galois Lattices, below only the first family of methods will be addressed. In particular, in the next subsection, within the existing work concerning the combination of domain ontologies and FCA techniques, one proposal concerning a similarity

measure for Concept Lattices will be recalled [6, 7]. Finally, in the second subsection, similarity methods do not involving FCA are mentioned.

### 5.1. Methods and similarity measures relying on FCA

Ontology merging, that consists in taking two or more source ontologies and returning a merged ontology based on the given sources, has been investigated in [46, 47]. In particular, in the mentioned papers the *FCA-merge* method has been proposed, that is based on Ganter and Wille's work on FCA and lattice exploration [25]. Given two or more source ontologies, one context is constructed for each of them, by applying natural language processing techniques. Once the contexts have been defined, they are joined and a pruned Concept Lattice is derived, that is manually explored and transformed into the merged ontology by a knowledge engineer. The engineer has to resolve possible conflicts and duplicates, but there is automatic support from the FCA-merge tool which aims at guiding and focusing the engineer's attention on specific parts of the construction process [48].

In [49], a method joining domain ontologies and Galois Graphs for knowledge discovering and data mining has been proposed. In particular, ontologies are used for enhancing keyword-based information retrieval, e.g., filtering the keywords describing a document. Galois Lattices can be used to detect correlations within the knowledge discovery process, and/or to build more concise and accurate domain ontologies.

However, in the above mentioned proposals, concept similarity is not addressed.

In [6, 7], fuzzy conceptual data analysis has been investigated. In the mentioned papers, fuzzy Concept Lattices have been introduced as a generalization of the theory of Wille, for the modeling of vague (non-crisp) extents and intents of concepts. In particular, in [6] the problem of the combination (aggregation and decomposition) of conceptual knowledge within fuzzy Concept Lattices has been addressed. In [7] an important problem related to FCA has been analyzed, i.e., the large number of concepts that can be extracted from data. This problem is generally addressed by using factorization of Concept Lattices and in [7], an algorithm for computing a factor lattice of a fuzzy Concept Lattice has been proposed. In particular, factorization is made by similarity, and a similarity measure for concepts of fuzzy Concept Lattices has been proposed. According to this method, that has been extensively presented in [5], similarity is first addressed at level of attributes and objects. For instance, in the case of attributes, two attributes $a_1$, $a_2$ are similar if they cannot be separated by any concept, i.e., if for each concept $c$, $a_1$ belongs to the intent of $c$ if and only if also $a_2$ belongs to the intent of $c$ (analogously in the case of objects).

The main difference between the Belohlávek's approach and the one proposed in this paper consists in the similarity of concept attributes, i.e., the intensional components of concepts. In fact, in this paper similarity of attributes is established by a panel of experts in the given domain (see, for instance, [38]), and on the basis of it, similarity of concept intents is computed independently of the related extents (the sets of objects). It is, in particular, evaluated according to a re-visitation of the maximum weighted matching problem in bipartite graphs. In other words, the similarity measure defined in [5] has mainly been conceived for Concept Lattices, therefore by taking into account that the intents and extents of concepts are strictly intertwined. The approach proposed in this paper is more oriented to deal with domain ontologies where, in general, the intensional components of concepts are emphasized and can be defined without the extensional components.

Just to mention other methods relying on FCA, we recall [29, 27], where FCA is used as the basis for a practical and well-founded methodology to semi-automatic ontology extraction and

design. In particular, in [27] an approach for the semi-automatic extraction of a taxonomy of concepts, and its transformation into Horn clauses, has been proposed.

Finally, it is worth recalling that text mining techniques are also used to enrich domain ontologies, in particular to discover both taxonomic and non-taxonomic knowledge (see for instance [35, 37]). The combination of FCA techniques and text mining techniques is a promising direction that is acquiring interest in the literature, see for instance [26].

### 5.2. Other Similarity Measures

Below the main differences among the approach presented in this paper and the majority of existing proposals, including [23] which has inspired this work, are mentioned.

An aspect that is worth recalling in comparing the different metrics proposed in the literature concerns hierarchically related entities (concepts). Within *Semantic nets* and logic-based *Knowledge Representation*, we recall [43], where a metric on the power set of nodes in a Semantic net has been proposed. In particular, the conceptual distance of concepts that are hierarchically related has been defined by considering the length of the shortest path connecting them. Furthermore, in [13] the *Semantic-Distance Metric* ($SDM$) has been defined, which is based on weighted paths. In particular, in that paper concepts are connected by hyperonym/hyponym and synonym links, but concepts with similarity degrees strictly lesser than one are not addressed. In [15, 19], a constant value (specifically 0.5) is associated with *any* pair of hierarchically related concepts, independently of the level of refinement of the concepts, i.e., the number of common attributes. In [23], in addition to the notion of *flat structural similarity*, that can be used to evaluate similarity of any pair of concepts, the notion of *hierarchical structural similarity* for hierarchically related concepts has been defined. Such a notion is based on the *extensional* aspect of inheritance, i.e., the possible distribution of objects along the hierarchy. In particular, it is related to the probability for an object (instance) of a more general concept to be an object of one of its specialized concepts, under specific assumptions. In this paper, since in FCA both the intensional and extensional aspects are explicitly represented in a concept, the notion of *flat structural similarity* has been revisited in order to address not only the intensional component but also the extensional component of a concept. For this reason, there was no need of defining an additional similarity metric for hierarchically related concepts.

Concerning the general notion of concept similarity, we did not adopt the popular *Dice*'s function, as for instance in [8, 15]. In fact, with respect to our approach, such a function introduces a simplification since concept similarity is evaluated on the basis of the number of similar concept components divided by the total number of concept components of the two concepts, without explicitly considering in the computation their similarity degree. Analogously, in [18] semantic relatedness (similarity) is based on the aggregation of the interconnections between concepts, that is, the more properties two concepts have in common, the more closely related they are.

In [11], general forms of distance metrics for the computation of similarity measures have been defined, although with more emphasis on the similarity between objects, rather than concepts. In [45], a richer set of distinguishing characteristics has been proposed, that includes both the intensional (classes) and extensional (tokens) levels. However, there are a number of limitations, such as the necessity that two concepts are at the same hierarchy level to be compared.

Finally, in [33] the problem of the construction and maintenance of ontology hierarchies has been addressed. In particular, a set of algorithms has been defined to construct sound ISA hierarchies, starting either from concepts (attributes) or from instances (objects). However,

rather than concept similarity, the focus of [33] is on the satisfaction of dependencies referred to as *existence constraints*, that is a problem related to our future work, as described in the next section.

## 6. Conclusion and Future Work

In this paper a proposal concerning similarity of concepts within Concept Lattices has been presented. Similarity reasoning is becoming a fundamental activity in the development of the Net Economy, to perform business transactions and e-commerce, and the Semantic Web, for ontology mapping and ontology integration. It is also growing in importance in different areas, such as component-based information development, integration of multiple heterogeneous information sources for mediation and data warehousing etc..

The starting point of the similarity method proposed in this paper is the definition of a context and the similarity relation of an existing domain ontology, as defined by a panel of experts in the given domain. Once the Concept Lattice of the given context has been defined, it is possible to compare concepts of the same context, but it is also possible to address different contexts and to evaluate the similarity of concepts belonging to different Concept Lattices.

As a future work, we are planning to extend the proposed results to another research problem addressed in FCA, i.e., the derivation of all the *attribute implications* of a context. An attribute implication of a context is a pair of subsets of attributes, say $X$, $Y$, for which $X' \subseteq Y'$, that is, each object having all attributes of $X$ has also all attributes of $Y$. This notion corresponds to that of attribute inheritance in Semantic nets, and the definition of a metric for *similar attribute implications* could be an interesting problem to analyze. Currently, we are also planning to implement the proposed approach by using conceptual clustering methods that allow the reduction of the size of Concept Lattices (see for instance the *iceberg* Concept Lattices [47]).

## References

[1] F.Baader, B.Sertkaya; *Applying Formal Concept Analysis to Description Logics*; International Conference on Formal Concept Analysis (ICFCA), pp.261-286, 2004.

[2] M.Bain; *Inductive Construction of Ontologies from Formal Concept Analysis*; Australian Conference on Artificial Intelligence, pp.88-99, 2003.

[3] C.Batini, M.Lenzerini, S.B.Navathe; *A Comparative Analysis of Methodologies for Database Schema Integration*; ACM Comput. Surv. 18(4), pp.323-364, 1986.

[4] C.Beeri, A.Formica, M.Missikoff; *Inheritance Hierarchy Design in Object-Oriented Databases*; Data & Knowledge Engineering (DKE), 30(3), pp.191-216, 1999.

[5] R.Belohlávek; *Similarity relations in concept lattices*; J. Log. Comput. 10(6), pp.823-845, 2000.

[6] R.Belohlávek; *Combination of knowledge in fuzzy concept lattices*; Int. Journal of Knowledge-Based Intelligent Engineering Systems 6(1), pp.9-14, 2002.

[7] R.Belohlávek, J.Dvorák, J.Outrata; *Fast factorization of concept lattices by similarity: solution and an open problem*; In Proc. of Concept Lattices and their Applications (CLA), V.Snásel, R.Belohlávek (Eds), Ostrava, Czech Republic, pp.47-57, 2004.

[8] S.Bergamaschi, S.Castano, S.De Capitani di Vimercati, S.Montanari, M.Vicini; *An Intelligent Approach to Information Integration*; in Formal Ontology in Information Systems, N.Guarino (Ed); IOS Press, Amsterdam, 1998.

[9] T.Berners-Lee et al.; *The Semantic Web*; Scientific American, May 2001.

[10] G.Birkoff; *Lattice Theory*, Amer. Math. Soc. Providence, R.I., 1967.

[11] G.Bisson; *Learning in FOL with a similarity measure*; Proc. of 10th National Conference on Artificial Intelligence, San Jose, CA, July 12-16, pp.82-87, The AAAI Press/The MIT Press, 1992.

[12] A.Borgida, J.Mylopoulos, H.K.T.Wong; *Generalization/Specialization as a Basis for Software Specification*; in "On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages", pp.87-117, Springer Verlag, 1984.

[13] M.Bright, A.Hurson, S.Pakzad; *Automated Resolution of Semantic Heterogeneity in Multidatabases*; ACM Transactions on Database Systems, 19(2), pp.212-253, 1994.

[14] D.Calvanese, G.De Giacomo, M.Lenzerini; *A Framework for Ontology Integration*; Proc. of Semantic Web Working Symposium (SWWS), pp.303-316, 2001.

[15] S.Castano, V.De Antonellis, M.G.Fugini, B.Pernici; *Conceptual Schema Analysis: Techniques and Applications*; ACM Transactions on Database Systems, 23(3), pp.286-332, 1998.

[16] P.Cimiano, A.Hotho, G.Stumme, J.Tane; *Conceptual Knowledge Processing with Formal Concept Analysis and Ontologies*; Int. Conference on Formal Concept Analysis (ICFCA), pp.189-207, 2004.

[17] W.W.Cohen; *Data Integration Using Similarity Joins and a Word-Based Information Representation Language*; ACM Transactions on Information Systems, 18(3), 288-321, 2000.

[18] A.Collins, E.Loftus; *A Spreading Activiation Theory on Semantic Processing*; Psychological Review, 82, pp.407-428, 1975.

[19] E.Damiani, A.Formica, M.G.Fugini, M.Missikoff, R.Pizzicannella; *Reusing Analysis Schemas in ODB Applications: a Chart Based Approach*; First East-European Symposium on Advances in Databases and Information Systems, St.Petersburg, Russia, September 2-5, pp.406-415, Nevsky Dialect, 1997.

[20] Y.Ding, D.Fensel, M.Klein, B.Omelayenko; *The semantic web: yet another hip?* Data & Knowledge Engineering 41(2-3), pp.205-227, 2002.

[21] J.Euzenat; *Evaluating ontology alignment methods*; Semantic Interoperability and Integration, Dagstuhl Seminar Proceedings 04391 IBFI, Germany, 2005.

[22] A.Formica, H.D.Groger, M.Missikoff; *Object-Oriented Database Schema Analysis and Inheritance Processing: a Graph-Theoretic Approach*; Data & Knowledge Engineering (DKE), 24(2), pp.157-181, 1997.

[23] A.Formica, M.Missikoff; *Concept Similarity in SymOntos: an Enterprise Ontology Management Tool*; The Computer Journal, 45(6), pp.583-594, 2002.

[24] Z.Galil; *Efficient algorithms for finding maximum matching in graphs*; ACM Computing Surveys, 18, pp.23-38, 1986.

[25] B.Ganter, R.Wille; *Formal Concept Analysis: Mathematical Foundations*; Springer, Berlin, 1999.

[26] F.H.Gatzemeier, O.Meyer; *Text Schema Mining Using Graphs and Formal Concept Analysis*; Proc. of the 10th Int. Conference on Conceptual Structures, (ICCS), pp.107-121, LNCS 2393, Springer-Verlag, London, UK, 2002.

[27] H.Haav; *A Semi-automatic Method to Ontology Design by Using FCA*; Proc. of Concept Lattices and their Applications (CLA), V.Snásel, R.Belohlávek (Eds), Ostrava, Czech Republic, 2004.

[28] F.Hakimpour, A.Geppert; *Resolving semantic heterogeneity in schema integration*; Int. Conference on Formal Ontology in Information Systems (FOIS), pp.297-308, 2001.

[29] S.Hwang, H.G.Kim, H.S.Yang; *A FCA-Based Ontology Construction for the Design of Class Hierarchy*; Int. Conference on Computational Science and its Applications (ICCSA) (3), pp.827-835, 2005.

[30] Y.Kalfoglou, S.Dasmahapatra, Y.Chen-Burger; *FCA in Knowledge Technologies: Experiences and Opportunities*; Int. Conference on Formal Concept Analysis (ICFCA), pp.252-260, 2004.

[31] Y.Kalfoglou, W.M.Schorlemmer; *Ontology Mapping: The State of the Art*; Semantic Interoperability and Integration, Dagstuhl Seminar Proceedings 04391 IBFI, Germany, 2005.

[32] M.Klein; *Combining and relating ontologies: an analysis of problems and solutions*; in WS on Ontologies and Information Sharing, A.Gomez-Perez et Al. (Eds), IJCAI'01, Seattle, USA, 2001.

[33] N.Lammari, E.Metais; *Building and maintaining ontologies: a set of algorithms*; Data & Knowledge Engineering 48(2), pp.155-176, 2004.

[34] J.Madhavan, P.A.Bernstein, P.Domingos, A.Y.Halevy; *Representing and Reasoning about Mappings between Domain Models*; National Conference on Artificial Intelligence (AAAI), Edmonton, Alberta, Canada, pp. 80-86, 2002.

[35] A.Maedche, S.Staab; *Discovering Conceptual Relations from Text*; European Conference on Artificial Intelligence (ECAI), Berlin, Germany, pp.321-325, 2000.

[36] M.Missikoff, F.Taglino; *SymOntoX: A Web-Ontology Tool for e-Business Domains*, Proc. of the 4th Int. Conference on Web Information Systems Engineering (WISE), Rome, December 1012, 2003.

[37] M.Missikoff, P.Velardi, P.Fabriani; *Text Mining Techniques to Automatically Enrich a Domain Ontology*, Applied Intelligence 18(3), pp.323-340, 2003.

[38] M.Missikoff, X.F.Wang; *A group decision system for collaborative ontology building*; Proc. of Int. Conference on Group Decision and Negociation, La Rochelle, France, June 4-7, pp.153-160, 2001.

[39] N.F.Noy, H.Stuckenschmidt; *Ontology Alignment: An annotated Bibliography*; Semantic Interoperability and Integration, Dagstuhl Seminar Proceedings 04391 IBFI, Germany, 2005.

[40] C.Parent, S.Spaccapietra; *Issues and Approaches of Database Integration*; Commun. ACM 41(5), pp.166-178, 1998.

[41] G.Pierra; *The PLIB ontology-based approach to data integration*; IFIP Congress Topical Sessions, pp.13-18, 2004.

[42] U.Priss; *Formal Concept Analysis in Information Science*; Annual Review of Information Science and Technology (ARIST), Preview Volume 40, 2006.

[43] R.Rada, H.Mili, E.Bicknell, M.Blettner; *Development and application of a metric on semantic nets*; IEEE Transactions on Systems, Man, and Cybernetics, 19(1), pp.17-30, 1989.

[44] E.Rahm, P.A.Bernstein; *A survey of approaches to automatic schema matching*, VLDB J. 10(4), pp.334-350, 2001.

[45] G.Spanoudakis, P.Constantopoulos; *Similarity for Analogical Software Reuse: A Computational Model*; Proc. of the Eleventh European Conference on Artificial Intelligence, Amsterdam, The Netherlands, John Wiley&Sons, New York, pp.18-22, 1994.

[46] G.Stumme, A.Maedche; *FCA-MERGE: Bottom-Up Merging of Ontologies*; Proc. of International Joint Conference on Artificial Intelligence (IJCAI), Seattle, USA, pp.225-234, 2001.

[47] G.Stumme, R.Taouil, Y.Bastide, N.Pasquier, L.Lakhal; *Computing iceberg concept lattices with Titanic*; Data & Knowledge Engineering 42(2), pp. 189-222, 2002.

[48] G.Stumme; *Ontology Merging with Formal Concept Analysis*; Semantic Interoperability and Integration, Dagstuhl Seminar Proceedings 04391 IBFI, Germany, 2005.

[49] L.Szathmary, A.Napoli; *Knowledge organisation and information retrieval using Galois lattices*, in "Workshop on Knowledge Management and Organizational Memories - 16th European Conference on Artificial Intelligence (ECAI), Valencia, Spain", R.Dieng-Kuntz, N.Matta (Eds), pp.73-78, 2004.

[50] P.Tonella; *Formal Concept Analysis in Software Engineering*; Int. Conference on Software Engineering (ICSE), pp.743-744, 2004.

[51] M.Uschold, M.Gruninger; *Ontologies: Principles, Methods and Applications*; The Knowledge Engineering Review, 11(2), 1996.

[52] H.Wache, T.Voegele, U.Visser, H.Stuckenschmidt, G.Schuster, H. Neumann, S.Huebner; *Ontology-Based Integration of Information - A Survey of Existing Approaches*; Proc. of the IJCAI Workshop on Ontologies and Information Sharing Seattle, USA, August 4-5, pp.108-118, 2001.

[53] R.Wille; *Restructuring lattice theory: an approach based on hierarchies of concepts*; Sym. on Ordered Sets, I.Rival (Ed), Reidel, Dordrecht, Boston, 1982.

[54] A.Yahia, L.Lakhal, R.Cicchetti, J.P.Bordat; *iO2 An Algorithmic Method for Building Inheritance Graphs in Object Database Design*; Proc. of Int. Conference on Conceptual Modeling (ER), Cottbus, Germany, October 1996.