

---

## An argumentation framework based on confidence degrees to combine ontology mapping approaches

---

C. Trojahn\* and P. Quaresma

Departamento de Informática,  
Universidade de Évora, Évora, Portugal  
E-mail: cassia@di.uevora.pt  
E-mail: pq@di.uevora.pt  
\*Corresponding author

R. Vieira

Faculdade de Informática,  
Pontifícia Universidade Católica do Rio Grande do Sul,  
Porto Alegre, Brazil  
E-mail: renata.vieira@puers.br

**Abstract:** Ontology mapping has a key importance for applications such as information retrieval, database integration, and agent-communication. This paper presents an Argumentation Framework, with confidence degrees associated to the arguments, to combine ontology mapping approaches. Our agents apply individual mapping algorithms and cooperate in order to exchange their local results (arguments). Based on their preferences and confidence of the arguments, the agents compute their preferred mapping sets. The arguments in such preferred sets are viewed as the set of globally acceptable arguments. The model is evaluated using a benchmark for ontology mapping. The results are promising especially what concerns precision.

**Keywords:** argumentation framework; ontology mapping; mapping approaches.

**Reference** to this paper should be made as follows: Trojahn, C., Quaresma, P. and Vieira, R. (2008) 'An argumentation framework based on confidence degrees to combine ontology mapping approaches', *Int. J. Metadata, Semantics and Ontologies*, Vol.

**Biographical notes:** Cássia Trojahn is a PhD student at Department of Informatics, University of Évora, Évora, Portugal. She is working in question answering systems, ontology mapping approaches, ontology translation, and argumentation models for ontology mapping.

Paulo Quaresma is an Associated Professor at Department of Informatics, University of Évora, Évora, Portugal. He works with information retrieval and natural language processing, specially question answering systems, ontologies, and text mining.

Renata Vieira is an Assistant Professor at Faculdade de Informática, Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, Brazil. Her main research interests are related to natural language processing, agent communication and ontologies.

---

### 1 Introduction

Ontologies provide a formal description of the objects and their relations in a domain. They have a key importance for applications such as information retrieval, database integration, peer-to-peer systems, e-commerce, or semantic web services. In information retrieval systems, the terms defined in an ontology are used as metadata to markup the Web's content and these semantic markups are semantic index terms for information retrieval, in order to improve the information retrieval (Jun-feng et al., 2005). Other typical application is question answering, where the data sources used to construct the answer are described using populated ontologies instead of databases. Tasks in

both these systems require support from more than one ontology for obvious reasons, and an ontology mapping is required. In a question answering system, our approach can be used to define mappings between the ontologies on the fly to describe data sources available to construct an answer. The ontologies are taken as input to the individual agents and the argumentation result can be used to search in such data sources.

Ontology mapping process takes two ontologies as input and determines as output correspondences between the semantically related entities of those ontologies. There are several mapping approaches related to different aspects of concepts similarity. In this paper, we focus in an approach based on argumentation to combine

ontology mapping methods. Different ontology mapping approaches are combined, as terms may be mapped by a measure of lexical similarity (Stoilos et al., 2005; Maedche and Staab, 2002), or they can be evaluated semantically, usually on the basis of semantic oriented linguistic resources, or considering the term positions in the ontology hierarchy (Hakimpour and Geppert, 2001). It is assumed that the approaches are complementary to each other and combining different ones reflect better solutions when compared to the solutions of the individual approaches.

We extend an argumentation framework, namely, Value-based Argumentation Framework (VAF) (Bench-Capon, 2003), in order to represent arguments with confidence degrees. The VAF allows to determine which arguments are acceptable, with respect to the different *audiences* represented by different agents. We then associate to each argument a confidence degree, representing how confident an agent is in the similarity of two ontology terms.

Our agents apply different mapping approaches and cooperate in order to exchange their local results (arguments). Next, based on their preferences and confidence of the arguments, the agents compute their preferred mapping sets. The arguments in such preferred sets are viewed as the set of globally acceptable arguments.

We evaluate our argumentation model using a benchmark for ontology mapping.<sup>1</sup> The results are promising especially what concerns precision. These tests demonstrate the application of our model for practical ontologies.

The paper has been divided into the following sections: Section 2 introduces ontologies and the state-of-art formalism used to represent them; Section 3 comments on ontology mapping approaches; Section 4 comments on argumentation framework; Section 5 presents our argumentation model; Section 6 presents one walk through example; Section 7 presents the evaluation of our model; Section 8 presents the final remarks and the future work.

## 2 Ontologies and OWL-Description Logic

The standard definition of ontology most used in the literature is from Gruber (1993), “an explicit specification of the conceptualisation of the domain”. From this definition Gangemi et al. (2005) point out that:

- the ontology makes things explicit – without an ontology many design assumptions may be implicit in the executable representation
- the ontology is supposed to be formal: the notions it captures are thus precise and unambiguous
- the ontology concerns some specific domain
- the ontology represents a conceptualisation – different people will conceptualise a domain differently according to experience, and their tasks in

the domain – and there is no a single ontology applicable to a domain.

Specifically, ontologies contain the types of *objects* in the domain; the *attributes* which these objects may have; the *relationships* which these objects may enter into; and the *values* that the attributes may have for particular types.

Although the terms ‘taxonomy’ and ‘ontology’ are used to describe sometimes the same kind of knowledge representation, there is a substantial difference. A taxonomy contains only ‘is-a’ relationship (classes and subclasses) whereas an ontology describes a domain completely, using other kinds of relationships, such as ‘part-whole’, and more complex descriptions, such as transitive and inverse properties.

A well-known formalism used to represent ontologies is Description Logics (DLs), a family of Knowledge Representation (KR) formalisms that represent the knowledge of an application domain (‘the world’) by first defining the relevant concepts of the domain (its terminology), and then using these concepts to specify properties of objects and individuals occurring in the domain (Baader et al., 2003).

A DL Knowledge Base (KB) comprises two components: TBox and ABox. TBox contains the *terminology*, which specifies the vocabulary of an application domain. ABox contains assertions about named individuals in terms of the TBox. The vocabulary consists of *concepts* and *roles*. *Concepts* denote set of individuals while *roles* denote binary relationship between individuals. Atomic concepts and roles can be used to build complex description of concepts and roles, using *constructors*. The language for building descriptions is a feature of different DLs, and different systems are distinguished by their description languages, i.e., the expressiveness of the language according with the *constructors* that they support.

OWL-DL ontology language is a variant of the SHOIN(D) (Baader et al., 2003) DL, which provides constructors for full negation, disjunction, a restricted form of existential quantification, and reasoning with concrete datatypes. OWL-DL is the state-of-the-art to represent expressive ontologies.

The set of SHOIN(D) concepts is defined by the following syntactic rules, where A is an atomic concept, R is a role name, d is a concrete domain,  $c_i$  are individuals, and  $n$  is a non-negative integer:

$$\begin{aligned} C &\rightarrow A \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists R.C \mid \forall R.C \mid \\ & n S \mid n S \mid \{a_1, \dots, a_n\} \mid n T \mid n T \mid \\ & \exists T_1, \dots, T_n.D \mid \forall T_1, \dots, T_n.D \\ D &\rightarrow d \mid \{c_1, \dots, c_n\} \end{aligned}$$

Here, we consider the semantics of a SHOIN(D) KB by Haase and Motik (2005), which transforms KB axioms into a first-order formula. Each atomic concept is mapped into a unary predicate and each role is mapped into a binary predicate.

### 3 Ontology mapping approaches

The previous work of Rahm and Bernstein (2001), Shvaiko (2004) and Shvaiko and Euzenat (2004) present a broad overview of the various approaches on automated ontology matching, classifying the mapping approaches in terms of input and techniques utilised in the mapping process. Kalfoglou and Schorlemmer (2003) present other style of ontology mapping classification that is based on frameworks, methods and tools, translators, mediators, etc. Choi et al. (2006) present the broad scope of ontology mapping, mapping categories (mapping between an integrated global ontology and local ontologies, mapping between local ontologies, and mapping on ontology merging and alignment), their characteristics, and a comprehensive overview of ontology mapping tools, systems, and related work. We point out that Kalfoglou and Schorlemmer (2003) and Choi et al. (2006) present a different style of ontology mapping classification from the one proposed by us, which is based on the classification of Rahm and Bernstein (2001), Shvaiko (2004) and Shvaiko and Euzenat (2004). We point out that a recent reference on ontology mapping is Euzenat and Shvaiko (2007), which presents a broad overview of the proposals for ontology mapping presented in the literature.

Rahm and Bernstein (2001) distinguishes between individual and combining matchers. Individual matchers comprise schema-based and instance-based, element and structure levels, linguistic and constrained-based matching techniques. Combining matchers comprise hybrid and composite matchers.

Based on this previous taxonomy, Shvaiko (2004) distinguishes between heuristic and formal techniques at schema-level; and implicit and explicit techniques at element-and structure-level. Shvaiko and Euzenat (2004) introduces new criterions which are based on:

- general properties of matching techniques, i.e., approximate and exact techniques
- interpretation of input information, i.e., syntactic, external, and semantic techniques at element and structure levels
- the kind of input information, i.e., terminological, structural, and semantic techniques.

Moreover, Giunchiglia and Shvaiko (2004) distinguishes between weak semantics and strong semantics element-level techniques. Weak semantics techniques are syntax-driven techniques (e.g., techniques which consider labels as strings, or analyse data types, or soundex of schema elements) while strong semantics techniques exploit, at the element level, the semantics of labels (e.g., based on the use of thesaurus).

As in Rahm and Bernstein (2001), we distinguish between individual and combining matchers. However, we divided the individual matchers on data level, ontology level, or context level, but we kept the combining matcher divided on hybrid or composite.

At the data level, data instances are used as input to the matching process. At the ontology level, the terms of the ontology structure and the hierarchy are taking into account. Then, as Rahm and Bernstein (2001), we distinguish between element-level matcher and structure level matcher. Finally, the ontology's application context can be used, i.e., how the ontology entities are used in some external context. This is specially interesting, for instance, to identify WordNet sense that must be considered to specific terms.

At the element-level we consider, according to Shvaiko and Euzenat (2004), semantic and external matchers. However, we replaced the syntactic by lexical and added a constraint-based matchers. We assume that the term 'syntactic' refers to morpho-syntactic categories of words (i.e., implicating some word annotation). We consider that the term 'lexical' is more appropriated to refer to the category of approaches based on string similarity.

The lexical approaches use metrics to compare string similarity. One well-known measure is the Levenshtein distance or edit distance (Levenshtein, 1966), which is given by the minimum number of operations (insertion, deletion, or substitution of a single character) needed to transform one string into another. Based on Levenshtein measure, Maedche and Staab (2002) proposes a lexical similarity measure for strings, the String Matching (SM), that considers the number of changes that must be made to change one string into the other and weighs the number of these changes against the length of the shortest string of these two. Other common metrics are: the Smith–Waterman (Smith and Waterman, 1981), which additionally uses an alphabet mapping to costs; and the Euzenat et al. (2004) which searches for the largest common substring.

Semantic matchers consider semantic relations between concepts to measure the similarity between them, usually on the basis of one thesaurus or similar semantic oriented linguistic resources. The well-known WordNet<sup>2</sup> database, a large repository of English items, has been used to provide these relations. This kind of mapping is complementary to the pure string similarity metrics. Cases where string metrics fail to identify high similarity between strings that represent completely different concepts are common. For example, for the words 'score' and 'store' the Levenshtein metric returns 0.68, which is a high metric if we consider that they represent very different concepts. On the other hand terms like 'student' and 'learner' are semantically similar although they are lexically distant from each other.

Constraint-based matchers are based on data types, value ranges, uniqueness, cardinalities, and other information constraints in the matching process. For example, the similarity between two terms can be based on the equivalence of data types and domains, of key characteristics (e.g., unique, primary, foreign), or relationship cardinality (e.g., 1 : 1 relationships) (Rahm and Bernstein, 2001).

Finally, at the element-level, we consider that external matchers consider some type of external information, such as user input or previous matching results.

Structural matchers use the ontology structure as input to the matching process (i.e., the positions of the terms in the ontology hierarchy are considered). Several approaches using this intuition have been proposed: super(sub)-concept rules consider that if super or sub concepts are the same, the actual concepts are similar to each other (Dieng and Hug, 1998; Ehrig and Sure, 2004); bounded path matching takes two paths with links between classes defined by the hierarchical relations, compare terms and their positions along these paths, and identify similar terms (see, for instance, Anchor-prompt algorithm Noy and Musen, 2001; Hovy, 1998); leaves-rules, where two non-leaf schema elements are structurally similar if their leaf sets are highly similar, even if their immediate children are not, see, for example Madhavan et al. (2001).

We also consider, as Rahm and Bernstein (2001), hybrid and composite matchers, at combining matcher level. Hybrid matchers use multiple matching criteria (e.g., name and type equality) within an integrated matcher; and composite matchers (which can use a manual or automatic process) combine multiple match results produced by different match algorithms. Our approach is an automatic composite matcher and then we add a cooperative approach at automatic level, which can be based on negotiation or argumentation. We point out that an automatic mapping approach can be also based on machine learning techniques, as presented by Doan et al. (2003), which combines multiple matchers using a learning approach.

Due to the complexity of the problem using only one approach is usually not satisfactory. These approaches are complementary to each other. Combining different approaches must reflect a better solution when compared to the solutions of individual approaches.

## 4 Argumentation framework

Our argumentation model is based on the VAFs (Bench-Capon, 2003), a development of the classical argument system of Dung (1995). First, we present the Dung's framework, upon which the VAF rely. Next, we present the VAF and our extended framework.

### 4.1 Classical argumentation framework

Dung (1995) defines an argumentation framework as follows.

**Definition 4.1.1:** An argumentation framework is a pair  $AF = (AR, attacks)$ , where  $AR$  is a set of arguments and  $attacks$  is a binary relation on  $AR$ , i.e.,  $attacks \subseteq AR \times AR$ . An  $attack(A, B)$  means that the argument  $A$  attacks the argument  $B$ . A set of arguments  $S$  attacks an argument  $B$  if  $B$  is attacked by an argument in  $S$ .

The key question about the framework is whether a given argument  $A$ ,  $A \in AR$ , should be accepted. One reasonable view is that an argument should be accepted only if every attack on it is rebutted by an accepted argument (Dung, 1995). This notion produces the following definitions:

**Definition 4.1.2:** An argument  $A \in AR$  is *acceptable* with respect to set arguments  $S(acceptable(A, S))$ , if  $(\forall x)(x \in AR) \wedge (attacks(x, A)) \longrightarrow (\exists y)(y \in S) \wedge attacks(y, x)$ .

**Definition 4.1.3:** A set  $S$  of arguments is *conflict-free* if  $\neg(\exists x)(\exists y)((x \in S) \wedge (y \in S) \wedge attacks(x, y))$ .

**Definition 4.1.4:** A conflict-free set of arguments  $S$  is *admissible* if  $(\forall x)(x \in S) \longrightarrow acceptable(x, S)$ .

**Definition 4.1.5:** A set of arguments  $S$  is a *preferred extension* if it is a maximal (with respect to inclusion set) admissible set of  $AR$ .

A *preferred extension* represent a consistent position within AF, which can defend itself against all attacks and which cannot be further extended without introducing a conflict.

The purpose of Bench-Capon (2003) in extending the AF is to allow associate arguments with the social values they advance. Then, the attack of one argument on another is evaluated to say whether or not it succeeds by comparing the strengths of the values advanced by the arguments concerned.

### 4.2 Value-based Argumentation Framework

In Dung's frameworks, attacks always succeed. However, in many domains, including the one under consideration, arguments lack this coercive force: they provide reasons which may be more or less persuasive (Laera et al., 2006). Moreover, their persuasiveness may vary according to their audience.

The VAF is able to distinguish attacks from successful attacks, those which defeat the attacked argument, with respect to an ordering on the values that are associated with the arguments. It allows accommodate different audiences with different interests and preferences.

**Definition 4.2.1:** A Value-based Argumentation Framework (VAF) is a 5-tuple  $VAF = (AR, attacks, V, val, P)$  where  $(AR, attacks)$  is an argumentation framework,  $V$  is a nonempty set of values,  $val$  is a function which maps from elements of  $AR$  to elements of  $V$  and  $P$  is a set of possible audiences. For each  $A \in AR$ ,  $val(A) \in V$ .

**Definition 4.2.2:** An Audience-specific Value Based Argumentation Framework (AVAF) is a 5-tuple  $VAF_a = (AR, attacks, V, val, valpref_a)$  where  $AR, attacks, V$  and  $val$  are as for a VAF,  $a$  is an audience and  $valpref_a$  is a preference relation (transitive, irreflexive and asymmetric)  $valpref_a \subseteq V \times V$ , reflecting the value preferences of audience  $a$ .  $valpref(v_1, v_2)$  means  $v_1$  is preferred to  $v_2$ .

**Definition 4.2.3:** An argument  $A \in AR$  *defeats<sub>a</sub>* (or *successfully attacks*) an argument  $B \in AR$  for audience  $a$  if and only if both *attacks* ( $A, B$ ) and not *valpref* ( $val(B), val(A)$ ).

An attack succeeds if both arguments relate to the same value, or if no preference value between the values has been defined.

**Definition 4.2.4:** An argument  $A \in AR$  is *acceptable* to audience  $a$  (*acceptable<sub>a</sub>*) with respect to set of arguments  $S$ , *acceptable<sub>a</sub>*( $A, S$ ) if  $(\forall x)((x \in AR \wedge \text{defeats}_a(x, A)) \rightarrow (\exists y)((y \in S) \wedge \text{defeats}_a(y, x)))$ .

**Definition 4.2.5:** A set  $S$  of arguments is *conflict-free* for audience  $a$  if  $(\forall x)(\forall y)((x \in S \wedge y \in S) \rightarrow (\neg \text{attacks}(x, y) \vee \text{valpref}(val(y), val(x)) \in \text{valpref}_a))$ .

**Definition 4.2.6:** A *conflict-free* set of arguments  $S$  for audience  $a$  is *admissible* for an audience  $a$  if  $(\forall x)(x \in S \rightarrow \text{acceptable}_a(x, S))$ .

**Definition 4.2.7:** A set of arguments  $S$  in the VAF is a *preferred extension* for audience  $a$  (*preferred<sub>a</sub>*) if it is a maximal (with respect to set inclusion) *admissible* for audience  $a$  of  $AR$ .

In order to determine the preferred extension with respect to a value ordering promoted by distinct audiences, (Bench-Capon, 2003) introduces the notion of *objective* and *subjective* acceptance.

**Definition 4.2.8:** An argument  $x \in AR$  is *subjectively* acceptable if and only if  $x$  appears in the preferred extension for some specific audiences but not all. An argument  $x \in AR$  is *objectively* acceptable if and only if,  $x$  appears in the preferred extension for every specific audience. An argument which is neither objectively nor subjectively acceptable is said to be *indefensible*.

### 4.3 Extended Value-based Argumentation Framework

We extend the VAF in order to represent arguments with confidence degrees. Two elements have been added to the VAF: a set with confidence degrees and a function which maps from arguments to confidence degrees. The confidence value represents the confidence that an individual agent has in some argument.

We assumed that the confidence degrees is a criteria which is necessary to represent the ontology mapping domain. Confidence degrees are important in the context of agent-based IR systems in distributed heterogeneous environments, such as the web.

**Definition 4.3.1:** An Extended Value-based Argumentation Framework (E-VAF) is a 7-tuple  $E\text{-VAF} = (AR, \text{attacks}, V, val, P, C, valC)$  where  $(AR, \text{attacks}, V, val, P)$  is a VAF,  $C$  is a nonempty set of values

representing the confidence degrees,  $valC$  is a function which maps from elements of  $AR$  to elements of  $C$ .  $valC \subseteq C \times C$  and  $valprefC(c_1, c_2)$  means  $c_1$  is preferred to  $c_2$ .

**Definition 4.3.2:** An argument  $x \in AR$  *defeats<sub>a</sub>* (or *successfully attacks*) an argument  $y \in AR$  for audience  $a$  if and only if  $\text{attacks}(x, y) \wedge (\text{valprefC}(valC(x), valC(y)) \vee (\neg \text{valpref}(val(y), val(x)) \wedge \neg \text{valprefC}(valC(y), valC(x))))$ .

An attack succeeds if (a) the confidence degree of the attacking argument is greater than the confidence degree of the argument being attacked; or if (b) the argument being attacked does not have greater preference value than attacking argument (or if both arguments relate to the same preference values) and the confidence degree of the argument being attacked is not greater than the attacking argument.

**Definition 4.3.3:** A set  $S$  of arguments is *conflict-free* for audience  $a$  if  $(\forall x)(\forall y)((x \in S \wedge y \in S) \rightarrow (\neg \text{attacks}(x, y) \vee (\neg \text{valprefC}(valC(x), valC(y)) \wedge (\text{valpref}(val(y), val(x)) \vee \text{valprefC}(valC(y), valC(x)))))$ .

## 5 E-VAF for ontology mapping

In our model, dedicated agents encapsulate different mapping approaches. Each approach represents a different audience in an E-VAF, i.e., the agents' preferences are based on specific approach used by the agent. Different audiences represent different preferences in the arguments. In this paper, we consider three audiences: lexical ( $L$ ), semantic ( $S$ ), and structural ( $E$ ) (i.e.,  $P = \{L, S, E\}$ , where  $P \in E\text{-VAF}$ ). We point out that our model is extensible to other audiences.

### 5.1 Argumentation generation

First, the agents work in an independent manner, applying the mapping approaches and generating mapping sets. The mapping result will consist of a set of all possible correspondences between terms of two ontologies. A mapping  $m$  can be described as a 3-tuple  $m = (t_1, t_2, R)$ , where  $t_1$  corresponds to a term in the ontology 1,  $t_2$  corresponds to a term in the ontology 2, and  $R$  is the mapping relation resulting from the mapping between these two terms. The lexical and semantic agents are able to return *equivalence* value to  $R$ , while the structural agents return *sub-class* or *super-class* values to  $R$ . Each mapping  $m$  is represented as an argument. Now, we can define arguments as follows:

**Definition 5.1.1:** An *argument*  $\in AR$  is a 4-tuple  $x = (m, a, c, h)$ , where  $m$  is a mapping;  $a \in P$  is the agent's audience generating that argument (agent's preference, i.e., lexical, semantic or structural);  $c \in C$  is the confidence degree associated to that mapping (*certainty*)

or *uncertainty*, as it will be commented below);  $h$  is one of  $\{-, +\}$  depending on whether the argument is that  $m$  does or does not hold.

The confidence degree is defined by the agent when applying the specific mapping approach. Here, we assumed  $C = \{\text{certainty}, \text{uncertainty}\}$ , where  $C \in \text{E-VAF}$ .

### 5.1.1 Lexical agent

The lexical agent adopts a metric to compare string similarity. We used the *lexical similarity* proposed by Maedche and Staab (2002). This metric is based on the Levenshtein distance (edit distance) (Levenshtein, 1966), which is given by the minimum number of operations (insertion, deletion, or substitution of a single character) needed to transform one string into another. The lengths of the compared terms are considered to compute the *lexical similarity*. This metric returns a value from the interval  $[0, 1]$ , where 1 indicates high similarity between two terms.

The agents are able to deal with compound terms. The first step in this process is the tokenisation. The terms are parsed into tokens by a tokeniser which also removes stop words ('and', 'of', etc). The *confidence* of an argument is computed according to the *lexical similarity* between each token of the two compared terms. Table 1 shows the possible values to  $s$  and  $h$ , where  $t_{S_n}$  correspond to some token of the source term (source ontology), and  $t_{T_n}$  correspond to some token of the target term (target ontology). We assume that two terms are *lexically similar* if the *lexical similarity* is greater than a *threshold*  $r$ .

**Table 1**  $h$  and  $c$  to lexical audience

$c$	$+(h)$
<i>Certainty</i>	$t_{S1}$ lexically similar to $t_{T1}$
<i>Uncertainty</i>	$t_{S1}$ lexically similar to some $a_{T1}, \dots, t_{Tn}$ Some $t_{S1}, \dots, t_{Sn}$ lexically similar to $t_T$ Some $t_{S1}, \dots, t_{Sn}$ lexically similar to some $t_{T1}, \dots, t_{Tn}$
$c$	$-(h)$
<i>Uncertainty</i>	Otherwise

When all tokens are *lexically similar* with each other, the terms match and the *confidence* of the argument is *certainty*. In this case, for instance, the lexical agent generates an argument  $x = (m, L, \text{certainty})$ , where  $m = (t_1, t_2, +)$ .

If *some* tokens of the terms are *lexically similar*, the *confidence* of the argument is *uncertainty*. If there are no lexically similar tokens between the terms, the agent is not sure that the terms map (i.e.,  $h$  equals to  $-$  and *confidence* equals to *uncertainty*), because this agent knows that other agent can resolve this mapping. In the specific case, if there is no lexical similarity between the terms, the semantic agent can resolve that mapping.

### 5.1.2 Semantic agent

The semantic agent considers the semantic (i.e., synonym, hyponym, and hypernym) relations between concepts

to measure the similarity between them, on the basis of WordNet<sup>3</sup> database, a large repository of English semantically related items. Table 2 shows the possible values to  $s$  and  $h$  according to the semantic similarity.

**Table 2**  $h$  and  $c$  to semantic audience

$c$	$+(h)$
<i>Certainty</i>	$t_{S1}$ semantic relation with $t_{T1}$
<i>Uncertainty</i>	$t_{S1}$ some semantic relation with some $t_{T1}, \dots, t_{Tn}$ Some $t_{S1}, \dots, t_{Sn}$ semantic relation with $t_T$ Some $t_{S1}, \dots, t_{Sn}$ semantic relation with some $t_{T1}, \dots, t_{Tn}$
$c$	$-(h)$
<i>Certainty</i>	Otherwise

When all tokens have semantic relation with each other, the *confidence* of the argument is *certainty*. The agent generates, for instance, an argument  $x = (m, S, \text{certainty})$ , where  $m = (t_1, t_2, +)$ . If *some* tokens have semantic relation, the *confidence* of the argument is *uncertainty*. Otherwise, if there are no semantic relation between the tokens, the agent is not sure that the terms map (i.e.,  $h$  equals to  $-$  and *confidence* equals to *uncertainty*), because this agent knows that other agent can resolve the mapping. In the specific case, when the searched terms are not available in WordNet, the lexical agent can decide the mapping. It is common because there is no complete lexical database for every domain (i.e., WordNet is incomplete for some domains).

### 5.1.3 Structural agent

The structural agent considers the positions of the terms in the ontology hierarchy to verify if the terms can be mapped. First, it is verified if the super-classes of the compared terms are lexically similar. If not, the semantic similarity between them is used. If the super-classes of the terms are lexically or semantically similar, the terms can be matched. The argument is generated according to the lexical or semantic comparison. For instance, if the super-classes of the terms are not lexically similar, but they are synonymous (semantic similarity), an argument  $x = (m, E, \text{confidence})$ , where  $m = (t_1, t_2, +)$ , is generated, where *confidence* varies according to the rules from Tables 1 or 2.

However, there are two main differences among the *confidences* returned by the structural, lexical and semantic agents.

If the structural agent does not find similarity (lexical or semantic) between the super-classes of the compared terms, it is because the terms can be not mapped (i.e., the terms occurs in different contexts). Then, the *confidence* for no mapping is *certainty*. Otherwise, if the structural finds similarity between the super-classes of the compared terms, it is because they can be mapped, but it does not mean that the terms are synonymous, then the *confidence* for the mapping is *uncertainty*. For instance, for the terms 'Publication/Topic' and 'Publication/Proceedings', the structural agent indicates

that the terms can be mapped because they have the same super-class, but not with *confidence* equals to *certainty* because it is no able to indicate that the terms are similar. Otherwise, for the terms ‘Digital-Camera/Accessories’ and ‘Computer/Accessories’, the agent can indicate that the terms can not be mapped because they occur in different contexts (*no-mapping* with *confidence* equals to *certainty*).

## 5.2 Preferred extension generation

After generating their set of arguments, the agents exchange with each other their arguments. When all agents have received the set of arguments of the each other, they generate their *attacks* set. An *attack* (or counter-argument) will arise when we have arguments for the mapping between the same terms, but with conflicting values of *h*. For instance, an argument  $x = (m_1, L, \textit{certainty}, +)$  have as an *attack* an argument  $y = (m_2, E, \textit{certainty}, -)$ , where  $m_1$  and  $m_2$  refer to the same terms in the ontologies. The argument  $y$  also represents an *attack* to the argument  $x$ .

When the set of arguments and attacks have been produced, the agents need to define which of them must be acceptable. To do this, the agents compute their preferred extension, according to the audiences and confidence degrees. A set of arguments is *globally subjectively acceptable* if each element appears in the preferred extension for some agent. A set of arguments is *globally objectively acceptable* if each element appears in the preferred extension for every agent. The arguments are neither objectively nor subjectively acceptable are considered *indefensible*.

## 6 Walk through example

Let us consider that three agents need to obtain a consensus about mappings that link corresponding class names in two different ontologies. We considered lexical (*L*), semantic (*S*), and structural (*E*) audiences (mapping approaches). We have selected the manual mappings between the ontologies described in Tables 3 and 4.

As shown in Table 5, the preferred extensions of the agents are composed by the arguments generated by the corresponding audience. The preferred extension of the lexical agents is {1, 5, 7}; the preferred extension of the semantic agent is {2, 5, 8}; and the preferred extension of the structural agent is {3, 6, 9}. Considering the ‘*subjectively acceptable*’ arguments, the arguments 1, 2, 3, 5, 6, 7, and 8 can be considered as consensus (note that the argument 4 is not acceptable).

**Table 3** DL ontology 1

Consumer-electronics $\sqsubseteq$ $\top$
Personal-computers $\sqsubseteq$ Consumer-electronics
Microprocessors $\sqsubseteq$ Personal-computers
Accessories $\sqsubseteq$ Personal-computers
Photo-and-cameras $\sqsubseteq$ Consumer-electronics

**Table 4** DL ontology 2

Electronics $\sqsubseteq$ $\top$
PC $\sqsubseteq$ Electronics
PC-board $\sqsubseteq$ PC
Cameras-and-photo $\sqsubseteq$ Electronics
Accessories $\sqsubseteq$ Cameras-and-photo
Digital-cameras $\sqsubseteq$ Cameras-and-photo

**Table 5** Arguments and attacks

ID	Argument	At.
1	(Photo-and-camera, Camera-and-photo, +, L, <i>certainty</i> )	–
2	(Photo-and-camera, Camera-and-photo, +, S, <i>certainty</i> )	–
3	(Photo-and-camera, Camera-and-photo, +, E, <i>uncertainty</i> )	–
4	(Personal-computer, PC, –, L, <i>uncertainty</i> )	5
5	(Personal-computer, PC, +, S, <i>certainty</i> )	4
6	(Personal-computer, PC, +, E, <i>uncertainty</i> )	4
7	(Consumer-electronic, Electronic, +, L, <i>uncertainty</i> )	–
8	(Consumer-electronic, Electronic, +, S, <i>uncertainty</i> )	–
9	(Consumer-electronic, Electronic, +, E, <i>uncertainty</i> )	–

## 7 Evaluation

The evaluation was made considering a benchmark for ontology mapping.<sup>4</sup> The set of ontologies we used are the following: the reference ontology is compared with itself (*Test 101*); the reference ontology is compared with another irrelevant ontology (*Test 102*); the reference ontology is compared with itself with labels replaced by synonyms (*Test 205*). The reference ontology used in *Test 101* has 97 terms (33 class names; 40 data attributes; 24 object attributes).

The agents were implemented in Java, and the experiments ran on Intel(R) Core(TM) Duo UCP 2.00 GHz, 2038 MB. JWordNet API,<sup>5</sup> which is an interface to the WordNet database, is required by the semantic agent.

Figure 1 shows the results for the *Test 101*, *Test 102*, and *Test 205*. For *Test 101*, 97 manual mappings had been defined. We considered only the mappings with *certainty* retrieved by the argumentation model. 97 mappings were retrieved. So, the *precision* and *recall* for this test is one (Figure 2). For *Test 102*, there is no manual mapping, considering that the source ontology is compared with other completely irrelevant. However, our model retrieved two mappings with *certainty*. We can not compute precision and recall when there are no reference manual mappings. For *Test 205*, 55 mappings were retrieved, from which one was incorrect. In this test, the precision was 0.98 and recall was 0.50 (Figure 2). The low recall is associated with some problems when searching in WordNet database:

some multiwords terms can not be retrieved (i.e., *start page* when compared with *beginning*); there is information that need to be retrieve using a thesaurus (i.e., *short name* when compared with *acronym*); some information could be retrieved by using the gloss of the synsets and at moment we do not use this information (i.e., the relationships between the terms *price* and *amount* is represented in the description of the term *price*: ‘the amount of money needed to purchase something’). So, we point out that better results can be obtained when considering these problems and using complementary databases such as thesaurus.

The results are promising especially what concerns precision. Moreover, using these tests, we demonstrated the application of our model for practical ontologies.

Figure 1 Results

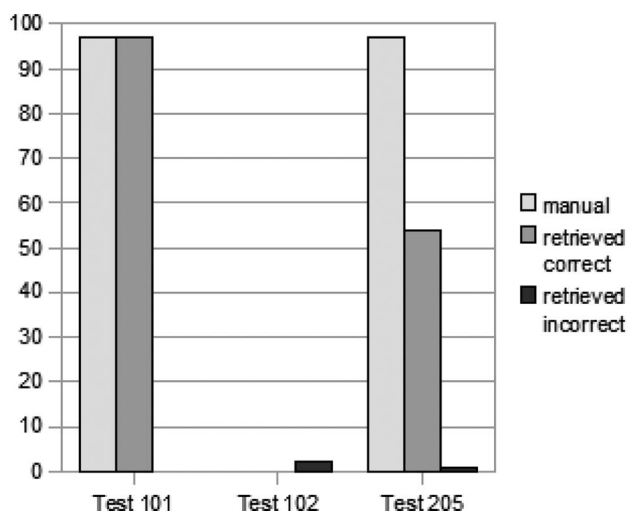
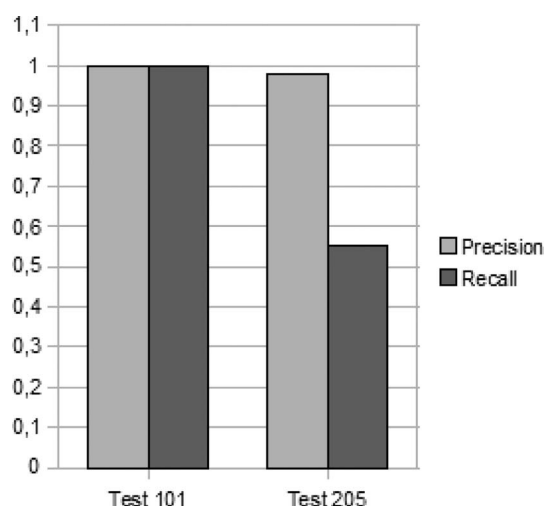


Figure 2 Precision and recall



## 8 Final remarks and future work

This paper presented a composite mapping approach based on the argumentation formalism to map ontologies. It was used an extended argumentation framework, namely VAF, with confidence degrees associated to arguments.

The VAF allows to determine which arguments are acceptable, with respect to the different preferences represented by different agents. The extension associates to each argument a confidence degree, representing the confidence that a specific agent has in that argument. We assumed that the confidence degrees is a criteria which is necessary to represent the ontology mapping domain. We have used different agents' output which use distinct mapping algorithms in order to verify the behaviour of our model.

We evaluated our argumentation model using a benchmark for ontology mapping.<sup>6</sup> The results were promising especially what concerns precision. Using these tests we demonstrated the application of our model for practical ontologies.

In the future, we intend to develop further tests considering also agents using constraint-based mapping approaches (i.e., the similarity between two terms can be based on the equivalence of data types and domains, of key characteristics, or relationship cardinality); use the ontology's application context in our mapping approach (i.e., how the ontology entities are used in some external context, which is especially interesting, for instance, to identify WordNet senses that must be considered to specific terms). Next, we will use the mapping result as input to an ontology translation process in a question answering system for the law domain.

## Acknowledgement

The first author is supported by the Programme Alban, the European Union Programme of High Level Scholarships for Latin America, scholarship No. E05D059374BR. This work has also partial support of the Brazilian research funding agency CNPq, projects 309096/2006-5 and 550852/2007-7.

## References

- Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D. and Patel-Schneider, P.F. (Eds.) (2003) *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press.
- Bench-Capon, T. (2003) 'Persuasion in practical argument using value-based argumentation frameworks', *Journal of Logic and Computation*, Vol. 13, pp.429–448.
- Choi, N., Song, I.-Y. and Han, H. (2006) 'A survey on ontology mapping', *SIGMOD Rec.*, Vol. 35, No. 3, pp.34–41.
- Dieng, R. and Hug, S. (1998) 'Comparison of personal ontologies represented through conceptual graphs', *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pp.341–345.
- Doan, A., Madhavan, J., Dhamankar, R., Domingos, P. and Helevy, A. (2003) 'Learning to match ontologies on the semantic web', *VLDB Journal (Special Issue on the Semantic Web)*, URL: [citeseer.ist.psu.edu/article/doan03learning.html](http://citeseer.ist.psu.edu/article/doan03learning.html)



- Dung, P. (1995) 'On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games', *Artificial Intelligence*, Vol. 77, pp.321–358.
- Ehrig, M. and Sure, Y. (2004) 'Ontology mapping – an integrated approach', *Proceedings of the European Semantic Web Symposium*, pp.76–91.
- Euzenat, J. and Shvaiko, P. (2007) *Ontology Matching*, Springer-Verlag, Heidelberg, DE.
- Euzenat, J., Le Bach, T., Barrasa, J., Bouquet, P., De Bo, J., Dieng-Kuntz, R., Ehrig, M., Hauswirth, M., Jarrar, M., Lara, R., Maynard, D., Napoli, A., Stamou, G., Stuckenschmidt, H., Shvaiko, P., Tessaris, S., Van Acker, S. and Zaihrayeu, I. (2004) *State of the Art on Ontology Alignment*, Technical Report.
- Gangemi, A., Pisanelli, D.M. and Steve, G. (2005) 'A formal ontology framework to represent norm dynamics', *Congreso Internacional de Culturas y Sistemas Juridicos Comparados*.
- Giunchiglia, F. and Shvaiko, P. (2004) 'Semantic matching', *Knowledge Engineering Review*, Vol. 18, No. 3, pp.265–280.
- Gruber, T.R. (1993) 'Towards principles for the design of ontologies used for knowledge sharing', in Guarino, N. and Poli, R. (Eds.): *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Kluwer Academic Publishers, Deventer, The Netherlands, URL: [citeseer.ist.psu.edu/gruber93toward.html](http://citeseer.ist.psu.edu/gruber93toward.html)
- Haase, P. and Motik, B. (2005) 'A mapping system for the integration of owl-dl ontologies', *IHIS*, pp.9–16.
- Hakimpour, F. and Geppert, A. (2001) 'Resolving semantic heterogeneity in schema integration: an ontology approach', *Proceedings of the International Conference on Formal Ontology in Informational Systems*, ACM, New York, NY, USA, pp.297–308.
- Hovy, E. (1998) 'Combining and standardizing large-scale, practical ontologies for machine translation and other uses', *Proceedings of the First International Conference on Language Resources and Evaluation*, pp.535–542.
- Jun-feng, S., Wei-ming, Z., Wei-dong, X., Guo-hui, L. and Zhen-ning, X. (2005) 'Ontology-based information retrieval model for the semantic web', *EEE '05: Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service, (EEE'05) on e-Technology, e-Commerce and e-Service*, IEEE Computer Society, Washington, DC, USA, pp.152–155.
- Kalfoglou, Y. and Schorlemmer, M. (2003) 'Ontology mapping: the state of the art', *The Knowledge Engineering Review*, Vol. 18, No. 1, pp.1–31.
- Laera, L., Tamma, V., Euzenat, J., Bench-Capon, T. and Payne, T.R. (2006) 'Reaching agreement over ontology alignments', *Proceedings of 5th International Semantic Web Conference (ISWC 2006)*, pp.371–384.
- Levenshtein, V. (1966) 'Binary codes capable of correcting deletions and insertions and reversals', *Soviet Physics Doklady*, Vol. 10, No. 8, pp.707–710.
- Madhavan, J., Bernstein, P. and Rahm, E. (2001) 'Generic schema matching with cupid', *Proceedings of the Very Large Data Bases Conference*, pp.49–58.
- Maedche, A. and Staab, S. (2002) 'Measuring similarity between ontologies', *Proceedings of the European Conference on Knowledge Acquisition and Management*, pp.251–263.
- Noy, N. and Musen, M. (2001) 'Anchor-prompt: using non-local context for semantic matching', *Proceedings of the Workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI)*, pp.63–70.
- Rahm, E. and Bernstein, P.A. (2001) 'A survey of approaches to automatic schema matching', *VLDB*, Vol. 10, pp.334–350.
- Shvaiko, P. (2004) *A Classification of Schema-Based Matching Approaches*, Technical Report, Informatica e Telecomunicazioni, University of Trento.
- Shvaiko, P. and Euzenat, J. (2004) *A Survey of Schema-Based Matching Approaches*, Technical Report, Informatica e Telecomunicazioni, University of Trento.
- Smith, T. and Waterman, M. (1981) 'Identification of common molecular subsequences', *Journal of Molecular Biology*, Vol. 147, pp.195–197.
- Stoilos, G., Stamou, G. and Kollias, S. (2005) 'A string metric for ontology alignment', *ISWC, 4th International Semantic Web Conference (ISWC 2005)*, pp.624–637, <http://www.image.ece.ntua.gr/publications.php>

## Notes

<sup>1</sup><http://oaei.ontologymatching.org/>

<sup>2</sup><http://www.wordnet.princeton.edu>

<sup>3</sup><http://www.wordnet.princeton.edu>

<sup>4</sup><http://oaei.ontologymatching.org/tests/>

<sup>5</sup><http://jwn.sourceforge.net> (using WordNet 2.1).

<sup>6</sup><http://oaei.ontologymatching.org/>