

Interoperability among Distributed Overlapping Ontologies – A Fuzzy Ontology Framework

Muhammad Abulaish
Member, IEEE & IEEE-CS
Department of Mathematics
Jamia Millia Islamia (A Central Univ.)
Jamia Nagar, New Delhi – 25, India
abulaish@computer.org

Lipika Dey
Member, IEEE & IEEE-CS
Department of Mathematics
Indian Institute of Technology
Hauz Khas, New Delhi – 16, India
lipika@maths.iitd.ernet.in

Abstract

Ontologies are proposed as a means for knowledge sharing among applications but, it is often not possible to converge to a single unambiguous ontology that is acceptable to all knowledge engineers. Different ontologies vary greatly in terms of the level of detail of their representations, as well as the nature of their underlying logical specifications. Interoperability among different ontologies becomes essential to gain from the power of the existing domain ontologies. In this paper we have proposed a fuzzy ontology framework in which a concept descriptor is represented as a fuzzy relation which encodes the degree of a property value using a fuzzy membership function. Other than concept descriptors, the semantic relations in the ontology like IS-A, HAS-PART etc. are also associated a strength of association. The strength of association between two concepts determines the “uniformity” with which these two concepts have been defined identically across different ontologies. The fuzzy ontology framework provides appropriate support for application integration by identifying the most likely location of a particular term in the ontology.

Keywords: Semantic Web, Ontology amalgamation, Ontology mapping, Ontology merging, Fuzzy ontology structure.

1. Introduction

The Semantic Web (SW) envisaged by Berners-Lee [10] has achieved a good standing within the last years. The SW is offering a very interesting possibility to semantic based Natural Language Processing (NLP) applications as it envisions formal semantic models available to assure interoperability between distributed software agents. But, the scenario offered by the SW is a complex and

heterogeneous one, where different and/or partially overlapping resources will coexist in a melting pot of distinct cultures, perspectives and representation approaches. The formal semantic models are generally represented in the form of ontology which represents domain knowledge in a structured form and is increasingly being accepted as the key technology wherein key concepts and their inter-relationships are stored to provide a shared and common understanding of a domain across applications [4].

As more and more people get involved, many individual ontologies are created. Though ontologies are proposed as a means for knowledge sharing among applications, interoperability among different ontologies becomes essential to gain from the power of the SW. It is often not possible to converge to a single unambiguous ontology that is acceptable to all knowledge engineers. The design differences are sometimes warranted by the applications and sometimes it may be due to sheer difference of opinion among knowledge engineers. Thus it is found that the same domain is usually represented by similar, though not exactly identical ontology structures. Different ontologies vary greatly in terms of the level of detail of their representations, as well as the nature of their underlying logical specifications. However, interoperability among different ontologies becomes essential to gain from the power of the existing domain ontologies. To attain ontological interoperability it is necessary to resolve the inconsistencies in concept descriptions and inter-concept relations present across multiple ontological structures that define the same domain. Ontology matching is an important research area and various methodologies for integrating knowledge from multiple ontologies, merging overlapping ontologies and also translating ontologies have been proposed in the literature [1, 2, 3, 5, 8, 9]. Some of the shortcomings of the existing ontology matching algorithms are [11]:

- The systems that perform ontology mapping are often either embedded in an integrated environment for

ontology editing or are attached to a specific formalism.

- In most cases mapping and merging are based on *heuristics* that mostly use syntactic clues to determine correspondence or equivalence between ontology concepts, but rarely use the meaning of those concepts, i.e., their semantics.

It appears from observations made by Kalfoglou and Schorlemmer [11] that a reasonable approach would be to accept the use of multiple ontologies across heterogeneous information systems and, while leaving the different underlying representations alone, analyze these different representations to assess and exploit the similarity among the entities specified in different ontologies. In this paper, we propose that this can be achieved through the following:

- All ontologies can be viewed as fuzzy ontologies with all relations accompanied by a fuzzy membership value that reflects its strength. The strength of an inter-concept relation reflects how consistently it has been used between the pair of concepts across ontologies.
- Every concept is associated to a new descriptor called *concept consistency*, which is assigned a numeric value. *Concept consistency* is a function of strength of all the relations associated to the concept.

Unlike other related work in handling overlapping ontologies we do not propose to merge or amalgamate ontologies, it is rather proposed that every application is allowed to work with their own ontologies with the fuzzy values simply providing an idea about the prospective dissimilarities in definitions with others. The proposed methodology exploits both syntactic and semantic clues to judge the similarities and dissimilarities among the different schemes.

The rest of the paper is organized as follows. We present a brief overview of some related works on ontology integration in section 2. Section 3 provides a mathematical view of the ontology structure. The proposed extension of the ontology structure into fuzzy ontology structure is given in section 4. The applicability of the proposed fuzzy ontology structure for resolving inconsistencies among distributed overlapping ontologies is explained in section 5. This section also describes in detail how any general ontology can be converted into a fuzzy ontology and thereafter the consistency of concept definitions can be derived. Finally, we conclude the paper in section 6.

2. Related work

In order to gain from the power of the existing domain ontologies, the task of ontology matching or establishing analogy among concept descriptions from multiple

ontologies have been proposed in the literature. In this section we present a brief overview of some of them.

Calvanese *et al.* [3] have proposed an Ontology Integration System (OIS) to facilitate the mapping between a global ontology and local ontologies. The OIS system can be applied in a situation where there are various local ontologies, developed independently from each other, and it is required to build an integrated, global ontology as a means for extracting information from the local ones. GLUE [1] is a system that employs machine learning techniques to find semantic mappings between concepts stored in distinct and autonomous ontologies. Given two distinct ontologies, the mapping discovery process between their concepts is based on the measure of similarity which is defined through the joint probability distribution. KAON [2] is an ontology and Semantic Web tool suite aimed at providing a platform for automatic integration of ontologies. The emphasis of this system is on ontology definition and on formal properties for correctness and completeness. Castano *et al.* [9] have proposed H-MATCH algorithm for dynamically matching distributed ontologies. Unlike the above approaches H-MATCH combines semantic affinity evaluation strategies to obtain a flexible and dynamic algorithm. The H-MATCH algorithm is able to discover the location of semantically related concepts to a target argument without requiring a complete description and matching procedure between independent ontologies. However, matching depends on finding matching sub-graphs in two parent ontologies and fails when such sub-graphs are not found.

Ontology merging is also proposed in literature for resolving the ontology heterogeneity problems. Protégé has a number of plug-ins, among others PROMPT, which is an algorithm for merging and aligning ontologies [8]. When merging two ontologies PROMPT creates a list of suggested operations. An operation can, for instance, be to merge two terms or to copy a term to the new ontology. The ontology editing tool Chimaera [5] also supports ontology merging. Chimaera merges two semantically identical terms from different ontologies so that they are referred to by the same name in the resulting ontology.

3. Ontology for structured representation of domain knowledge – a mathematical view

An ontology can be viewed as a model of a domain that defines the concepts existing in that domain, their properties and the relationships between them and is typically represented as a knowledge base. For example, plant ontology specifies structural organization of plants in terms of parts and sub-parts like *stems*, *leaves*, *cells* etc.; the various categories of plants like *algae*, *legumes*, *ferns* and so on.

An ontology (Θ) organizes domain knowledge in terms of concepts (C), properties (P) and relations (\mathfrak{R}) and can be formally defined as follows.

Definition (Ontology) – An Ontology Θ is a triplet of the form

$\Theta = (C, P, \mathfrak{R})$, where:

- C is a set of concepts defined for the domain.
- P is a set of concept properties. A property $p \in P$ is defined as an instance of a ternary relation of the form $p(c, v, f)$, where $c \in C$ is an ontology concept, ‘ v ’ is a property value associated with ‘ c ’ and ‘ f ’ defines restriction facets on v . Some of the restriction facets are – *type* (f_t), *cardinality* (f_c), and *range* (f_r). The *type* facet f_t may be any one from the standard data types supported by ontology editors i.e., $f_t \in \{\text{Boolean, integer, float, string, symbol, instance, class, ...}\}$. The *cardinality* facet f_c defines the upper and lower limits on the number of values for the property. The *range* facet f_r specifies a range of values that can be assigned to the property.
- $\mathfrak{R} \subseteq C \times C \times R_T$ is a set of binary semantic relations defined between concepts in Θ . $R_T = \{\text{one-to-one, one-to-many, many-to-many}\}$ is the set of relation type.

\mathfrak{R} is recursively defined as follows:

- a. A set of atomic relations is defined as $\mathfrak{R}_a = \{\approx, \uparrow, \downarrow, \nabla, \Delta\}$ which have the following interpretations:

For any two ontological concepts $C_i, C_j \in C$

- * \approx denotes the equivalence relation. $C_i \approx C_j \Rightarrow C_i$ is equivalent to C_j . The *synonym* relation of natural language is modeled in an ontology using the equivalence relation. For example, through WordNet [6] we obtain that the word “*inn*” is synonymous to “*hotel*”. Some domain experts use either double arrowed line or line segment, labeled with “*same-as*” to represent equivalence relation in an ontology. If two concepts C_i and C_j are declared equivalent in an ontology then instances of concept C_i can also be inferred as instances of C_j and vice-versa.
- * \uparrow denotes the generalization relation. $C_i \uparrow C_j \Rightarrow C_i$ is a generalization of C_j . When an ontology specifies that C_i is a generalization of C_j , then C_j inherits all property descriptors associated with C_i , and these need not be repeated for C_j while specifying the ontology. \downarrow is the inverse of \uparrow . Hence, $C_i \uparrow C_j \Rightarrow C_j \downarrow C_i$, i.e., C_i is a generalization of C_j implies that C_j is a specialization of C_i . The relations \uparrow and \downarrow correspond to the semantic relations

“*hypernym*” and “*hyponym*” respectively. These relations are usually denoted by an arrow super scribed with “*is-a*” or “*kind-of*”, where the arrow is directed from the specialized class to the generalized class. Ontologies can also accommodate multiple inheritance, whereby a concept can acquire properties through multiple paths of specialization.

- * $C_i \nabla C_j \Rightarrow C_i$ has part C_j . Δ is inverse of ∇ . Hence $C_i \Delta C_j \Rightarrow C_i$ is a part of C_j . In an ontology, a concept which is defined as aggregation of other concepts is expressed using the relation ∇ . The “*has-part*” relation is equivalent to the “*holonym*” relation of WordNet.

- b. If $\mathfrak{R}_1, \mathfrak{R}_2 \in \mathfrak{R}$ be any two relations defined between concept-pairs in Θ and \circ denotes a composition operation, $\mathfrak{R}_1 \circ \mathfrak{R}_2$ is a valid relation.

4. Enhancement of ontology structure into fuzzy ontology structure

Traditionally concepts are described in an ontology using a $\langle \text{property, value, constraints} \rangle$ framework. The fuzzy ontology structure is created as an extension to the standard ontology structure. In the proposed design of a fuzzy ontology, a concept descriptor is represented as a fuzzy relation which encodes the degree of a property value using a fuzzy membership function. The proposed fuzzy ontology structure stores concept descriptions in a $\langle \text{property, value, qualifier, constraints} \rangle$ framework, where the value and the qualifier are both defined as a fuzzy set. This framework allows defining the property-value of a concept with differing degrees of fuzziness, without actually changing the concept description paradigm. Such concept descriptions can be termed as imprecise concept descriptions. Other than concept descriptors, other relations in the ontology like IS-A, HAS-PART etc. are also associated a strength of association. Mathematically, a fuzzy ontology (Θ_F) can be defined as follows.

Definition (Fuzzy Ontology) – A Fuzzy Ontology, Θ_F , is a quadruple of the form

$\Theta_F = (C, P_F, \mathfrak{R}_F, M)$, where:

- C has same interpretation as mentioned in section 3.
- P_F is a set of fuzzy concept properties. A property $p_f \in P_F$ is defined as a quadruple of the form $p_f(c, v_f, q_f, f)$, where $c \in C$ is an ontology concept, ‘ v_f ’ represents fuzzy attribute values and could be either *fuzzy numbers* or *fuzzy quantifiers*, ‘ q_f ’ models linguistic qualifiers and are *hedges*, which can control or alter the strength of an attribute value and f is the restriction facets on v_f .

- \mathfrak{R}_F is a set of inter-concept relations between concepts. Like fuzzy concept properties, \mathfrak{R}_F is defined as a quadruple of the form $\mathfrak{R}_F(c, c, t, q_f)$, where $c \in C$ is an ontology concept, 't' represents relation type, and 'q_f' models relation strengths and are linguistic variables, which can represent the strength of association between concept-pairs $\langle c, c \rangle$.
- The choice of *fuzzy numbers* or *fuzzy quantifiers* for values is dictated by the nature of the underlying attribute and also its restriction facets. The complete range of values over which an attribute can take values defines the *universe of discourse M*. The universe of discourse is decomposed into a collection of fuzzy sets. Each fuzzy set is defined over a *domain* that overlays part of the universe of discourse.

An interesting aspect of modeling attributes as fuzzy sets is that with an underlying set of numeric values, one can associate different fuzzy quantifier sets to represent different aspects of the same attribute. For example, a single price value can be interpreted as being “close to” or “far away” from another value of price, and at the same time can also be interpreted as “cheap” or “expensive.” Moreover, hedges can also be applied to create new fuzzy sets with different meanings. Thus modeling an attribute as a fuzzy set allows a single attribute to contribute to different types of imprecision in concept description. Applicability of this fuzzy ontology structure in retrieving and curating information from text documents have been thoroughly experimented and presented in [7]. In our application, the fuzzy membership function is determined

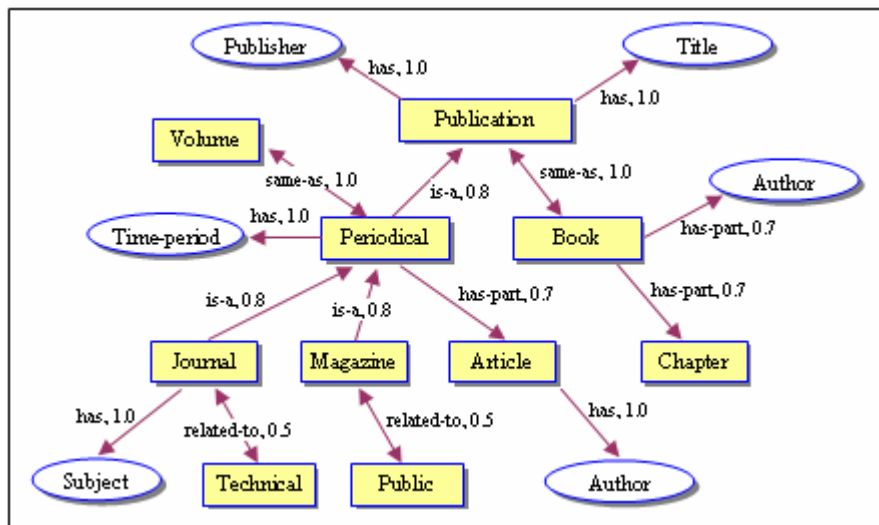


Figure 1. Ont1: an ontology derived through WordNet to represent *publication* domain

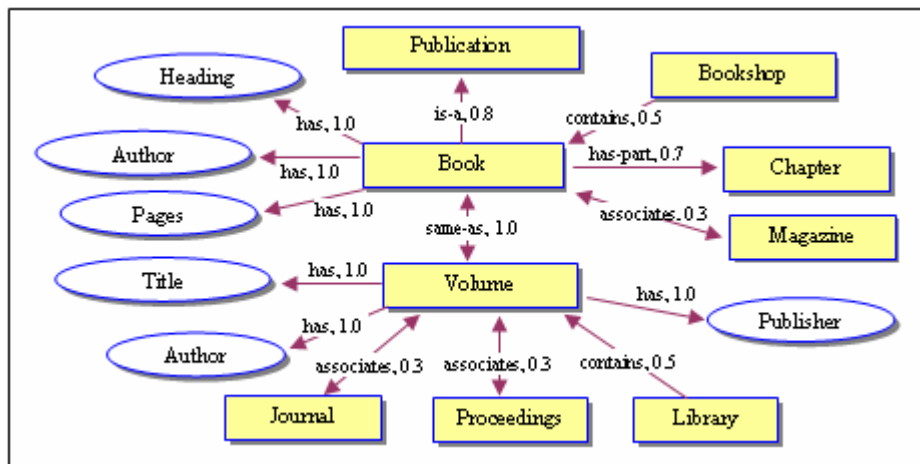


Figure 2. Ont2: an ontology taken from [10] to represent *publication* domain

through text mining. In this paper we have shown how the fuzzy ontology structure can be used to represent inconsistencies in concept descriptions among various overlapping ontologies representing the same domain.

5. Fuzzy ontology structures for representing inconsistent concept descriptions

We first illustrate the idea of inconsistent definition through an example and then go on to present how a measure of *concept consistency* is derived. Figures 1 and 2 show two different ontologies Ont1 and Ont2 respectively that describe the concept *publication* and other concepts related to it. In these figures, rectangles and ovals are used to represent concept and concept descriptors respectively. The ontology Ont1 has been hand-crafted following the inter-concept relations as apparent in WordNet 2.0. The ontology Ont2 has been taken from [9]. Several interesting observations can be made about these ontologies:

- While in Ont1 “Book” is described as synonymous to “Publication”, in Ont2 “Book” is described as a specialization of “Publication”. This means that in the earlier case, all instances of “Book” can be interpreted as “Publication” and vice-versa, though it is not so by the second ontology.
Many applications would tend to disagree with the first ontology, since publications like journals, magazines etc. are substantially different in nature from books, where books are usually one-time publications where as journals and magazines are periodicals. This is how Ont2 actually encodes the knowledge.
- It is observed that in Ont1, the descriptor “Volume” is not applicable for describing a “book”, while in Ont2 a volume is defined as an equivalent concept to the concept “book”.
Given the way the two ontologies define the relationship between “book” and “publication” this difference is bound to come up.
- “Pages” is used as a property descriptor in Ont2; however “pages” was not obtained as a *meronym* when we fed “book” to WordNet 2.0.
While in the earlier cases, there was no doubt about the need of the concepts in the ontology, this difference raises question about the necessity of using “Pages” as a concept descriptor while defining publications. There are several other such discrepancies.

Castano *et al.* [9] have suggested that the affinity between two concepts is a function of the relation that binds them. They have divided the relations into two categories

linguistic and *semantic*, and assigned numeric weights to these relations. These weights are shown in Table 1. The weights associated with the linguistic relationships are taken from ARTEMIS¹, where they have been tested on several real integration cases. The weights associated with semantic relations have been defined in HELIOS to express a measure of the strength of the concept connection posed by each relation for semantic affinity evaluation purposes. The higher is the weight associated with a semantic relation, the higher is the strength of the semantic connection between concepts.

To begin with, an ontology can be converted into a fuzzy ontology, as defined in section 4, where any relation is a fuzzy relation accompanied by its weight as defined in Table 1. The numeric values shown in figures 1 and 2 show fuzzy version of the ontologies Ont1 and Ont2 respectively, in which every relation is associated with its *linguistic* or *semantic* weight. These weights are used to define concept consistencies as explained in the next sub-section.

Table 1. Weights associated with linguistic and semantic relations

| Interpretation | Relation Name | Weight |
|----------------|-------------------|--------|
| Linguistic | Synonym | 1.0 |
| | Hypernym/ Holonym | 0.8 |
| | Related term | 0.5 |
| Semantic | Property | 1.0 |
| | Same as | 1.0 |
| | Kind-of/ is-a | 0.8 |
| | Part-of/ Has-part | 0.7 |
| | Contains | 0.5 |
| | Associates | 0.3 |

5.1. Computing concept consistencies

Concept consistency among multiple ontologies is computed as function of concept affinity. Castano *et al.* [9] have defined concept affinity between two concepts in an ontology as a function of relations that binds them in the ontology. The aim of the concept affinity is to evaluate the overall degree of relatedness of a related concept-pair $\langle C_i, C_j \rangle$ within an ontology. Since a related concept-pair $\langle C_i, C_j \rangle$ can have different relation paths between C_i and C_j , the concept affinity $CA(C_i, C_j)$ is equal to the highest-weight path of relationships between them in the ontology, i.e., for a given set of paths, P , between concepts C_i and C_j the concept affinity between them is calculated by using equation 1, in which the weight of a path is computed by multiplying the

¹ <http://islab.dico.unimi.it/artemis/d2i/>

weights of all relationships forming the path. In case, if no path exists between a pair of concept, the concept affinity is assumed to be zero.

$$CA(C_i, C_j) = \text{MAX}_{p \in P} \{ \text{Weight}(p) \} \dots \dots \dots (1)$$

Given N ontologies, for each ontology O^i , where $i = 1, 2, \dots, N$, concept affinity can be computed between every pair of concepts defined in that ontology. Concept affinity values calculated for the related concept-pairs in Ont1 and Ont2 are shown in tables 2 and 3 respectively. The rows in each table have been restricted to represent concept names only. The consistency of definition of a single concept C_i denoted by $\zeta(C_i)$, is dependent on its relationship with other concepts, where the domain of concepts vary over all ontologies. The dependence of $\zeta(C_i)$ on another concept C_j occurs in two different ways:

- (i) C_i co-occurs with C_j in multiple ontologies and is linked through same or different relations.
- (ii) C_i co-occurs with C_j in only one ontology

For every concept C_j , related to C_i , we compute the consistency of relationship between C_i and C_j , denoted by $\gamma(C_i, C_j)$, as follows:

- If C_j co-occurs with C_i in more than one ontologies, then $\gamma(C_i, C_j)$ is given by equation 2. This quantity computes a value that reflects the consistency that is observed in the definition of the inter-concept relationship between C_i and C_j across ontologies. For example, if C_j is related to C_i by a part-of relation in more than one ontology, and no other relationship between C_i and C_j is observed, then $\gamma(C_i, C_j) = 0.7$, which is the semantic weight for this relation. For concept C_i , let M_{ij} denote the set of concepts C_j with which it co-occurs in multiple ontologies.
- If C_j co-occurs with C_i in a single ontology then $\gamma(C_i, C_j)$ is given by equation 3. If a concept pair relationship is observed only once in the presence of multiple domain ontologies, the role of this equation is

to reduce the overall importance of this relationship. For concept C_i , let S_{ij} denote the set of concepts which co-occur with C_i only once, i.e., in a single ontology.

Finally, the overall consistency of concept C_i is computed by using equation 4. On applying the above definitions to the ontologies Ont1 and Ont2, the concept consistency values computed for the various concepts present in these ontologies are shown in table 4.

$$\gamma(C_i, C_j) = \frac{N \times \sum_{p=1}^{N-1} \sum_{q=p+1}^N CA_p(C_i, C_j) \times CA_q(C_i, C_j)}{\binom{N}{2} \times \sum_{p=1}^N CA_p(C_i, C_j)} \dots \dots \dots (2)$$

$$\gamma(C_i, C_j) = \frac{CA(C_i, C_j)}{n} \dots \dots \dots (3)$$

$$\zeta(C_i) = \frac{1}{|M_{ij}| + |S_{ij}|} \left[\sum_{C_j \in M_{ij}} \gamma(C_i, C_j) + \sum_{C_j \in S_{ij}} \gamma(C_i, C_j) \right] \dots (4)$$

It may be observed that these values are unique and does not change in association to an ontology. The overall role of the consistency value is to convey to the user about how consistently a concept has been described across multiple ontologies. It is observed from table 4 that the concept “Book” is defined fairly consistently across the two ontologies, though there is a lot of disagreement about the concept of a “Publication”. The most consistently defined concept is “Chapter” which in both the cases is defined as a “part-of” a Book.

The proposed method of using a fuzzy ontology attempts to quantify the consistency in the definition of multiple domain ontologies. Following Kalfoglou1 and Schorlemmer [11] principle, this method does not tend to integrate multiple ontologies rather produces a unique measure of consistency for each concept that is defined for any ontology. Thus every application is free to pursue its own ontology. However, having a global idea allows users to reconsider their ontologies if they so wish.

Table 2. Concept affinities between related concept-pairs in Ont1

| Concepts & Properties → Concepts ↓ | Public ation | Publis her | Title | Volume | Periodical | Time Period | Book | Author | Journal | Magaz ine | Article | Chapter | Subject | Technical | Public |
|---------------------------------------|-----------------|---------------|-------|--------|------------|----------------|------|--------|---------|--------------|---------|---------|---------|-----------|--------|
| Publication | 1.00 | 1.00 | 1.00 | 0 | 0 | 0 | 1.00 | 1.00 | 0 | 0 | 0 | 0.70 | 0 | 0 | 0 |
| Volume | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0 | 0 | 0.70 | 0.70 | 0 | 0 | 0 |
| Periodical | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0 | 0 | 0.70 | 0 | 0 | 0 | 0 |
| Book | 1.00 | 1.00 | 1.00 | 0 | 0 | 0 | 1.00 | 1.00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Journal | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0 | 0.70 | 0.70 | 1.00 | 0.50 | 0 |
| Magazine | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0 | 1.00 | 0.70 | 0.70 | 0 | 0 | 0.50 |
| Article | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.00 | 0 | 0 | 1.00 | 0 | 0 | 0 | 0 |
| Chapter | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.00 | 0 | 0 | 0 |
| Technical | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0 | 0.35 | 0.35 | 0.50 | 1.00 | 0 |
| Public | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0 | 0.50 | 0.35 | 0.35 | 0.50 | 0 | 1.00 |

Table 3. Concept affinities between related concept-pairs in Ont2

| Concepts & Properties → Concepts ↓ | Publication | Book | Bookshop | Chapter | Magazine | Heading | Author | Page | Volume | Publisher | Library | Proceedings | Journal | Title |
|---------------------------------------|-------------|------|----------|---------|----------|---------|--------|------|--------|-----------|---------|-------------|---------|-------|
| Publication | 1.00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Book | 1.00 | 1.00 | 0 | 0.70 | 0.30 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0 | 0.30 | 0.30 | 1.00 |
| Bookshop | 0 | 0.50 | 1.00 | 0.35 | 0.15 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 0 | 0.15 | 0.15 | 0.50 |
| Chapter | 0 | 0 | 0 | 1.00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Magazine | 0.30 | 0.30 | 0 | 0.21 | 1.00 | 0.30 | 0.30 | 0.30 | 0.30 | 0.30 | 0 | 0.09 | 0.09 | 0.30 |
| Volume | 1.00 | 1.00 | 0 | 0.70 | 0.30 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0 | 0.30 | 0.30 | 1.00 |
| Library | 0.50 | 0.50 | 0 | 0.35 | 0.15 | 0.50 | 0.50 | 0.50 | 0.50 | 0.50 | 1.00 | 0.15 | 0.15 | 0.50 |
| Proceedings | 0.30 | 0.30 | 0 | 0.21 | 0.09 | 0.30 | 0.30 | 0.30 | 0.30 | 0.30 | 0 | 1.00 | 0.09 | 0.30 |
| Journal | 0.30 | 0.30 | 0 | 0.21 | 0.09 | 0.30 | 0.30 | 0.30 | 0.30 | 0.30 | 0 | 0.09 | 1.00 | 0.30 |

Table 4. Concept consistency values for the concepts in Ont1 and Ont2

| Concept | Consistency Value | Concept | Consistency Value |
|-------------|-------------------|----------|-------------------|
| Publication | 0.39 | Volume | 0.63 |
| Periodical | 0.47 | Book | 0.64 |
| Journal | 0.39 | Magazine | 0.38 |
| Article | 0.50 | Chapter | 1.00 |
| Technical | 0.26 | Public | 0.26 |
| Bookshop | 0.22 | Library | 0.22 |
| Proceedings | 0.16 | | |

For all concepts defined as a part of the application ontology, the consistency values are known. The user gets an idea about the concepts that are not a part of their ontology. This can also be used as boost for those parts of the ontology that have high consistency values.

6. Conclusion

In this paper we have proposed a fuzzy ontology generation framework to create a fuzzy ontology structure through enhancement of the existing ontological structures to accommodate imprecision in concept descriptions and inter-concept relations. Besides other applications, reported in [7], the use of the proposed framework is shown to quantify inconsistencies in concept definitions across multiple overlapping ontologies representing the same domain. The proposed idea for ontology integration and amalgamation is different from others because this method does not tend to integrate or merge multiple ontologies rather produces a unique measure of consistency for each concept that is defined for any ontology. Thus every application is free to pursue its own ontology. However, having a global idea allows users to reconsider their ontologies if they so wish.

References

- [1] A. Doan, J. Madhavan, P. Domingos, and A. Halevy, "Learning to Map between Ontologies on the Semantic Web", in: *Proceedings of the 11th International World Wide Web Conference (WWW'02)*, Honolulu, Hawaii, USA, 2002.
- [2] B. Motik, A. Maedche, and R. Volz, "A Conceptual Modeling Approach for Semantics-driven Enterprise Applications", in: *Proceedings of the 1st International Conference on Ontologies, Databases and Application of Semantics*, 2002.
- [3] D. Calvanese, G.D. Giacomo, and M. Lenzerini, "A Framework for Ontology Integration", in: *Proceedings of the 1st Semantic Web Working Symposium*, 2001, pp. 303-316.
- [4] D. Fensel, F. van Harmelen, I. Horrocks, D.L. McGuinness, and P.F. Patel-Schneider, "OIL: An ontology Infrastructure for the Semantic Web", *IEEE Intelligent Systems*, 16 (2), 2001, pp. 38-45.
- [5] D. L. McGuinness, R. Fikes, J. Rice, and S. Wilder, "An Environment for Merging and Testing Large Ontologies", in: *Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning, Breckenridge, CO, USA, 2000*, pp. 483-493.
- [6] G. Miller, "WordNet: An Online Lexical Database", *International Journal of Lexicography*, 3(4), 1997.
- [7] M. Abulaish, and L. Dey, "Information Extraction and Imprecise Query Answering from Web Documents", *Web Intelligence & Agent Systems – An International Journal*, Vol. 4, No. 4, 2006 (to appear).
- [8] N.F. Noy, and M. Musen, "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment", in: *Proceedings of 17th National Conference on Artificial Intelligence*, Austin, USA, 2000, pp. 450-455.
- [9] S. Castano, A. Ferrara, and S. Montanelli, "H-MATCH: an Algorithm for Dynamically Matching Ontologies in Peer-based Systems", in: *Proceedings of the 1st International Workshop on Semantic Web and Databases (SWDB'03)*, 2003, pp. 231-250.
- [10] T. Berners-Lee, "Semantic Web Road Map, W3C Design Issues", 1998, <http://www.w3.org/DesignIssues/Semantic.html>.
- [11] Y. Kalfoglou, and M. Schorlemmer, "IF-Map: An Ontology-Mapping Method based on Information-Flow Theory", *Journal of Data Semantics*, 1(1), 2003, pp. 98-127.