

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSI*net*

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Ranking Ontologies Based on OWL Language Constructs

¹V. Ravi Sankar, ²A. Damodaram and ³P. Radha Krishna
¹Aurora's Engineering College, Bhongir Andhra Pradesh, India
²Jawaharlal Nehru Technological University, Kukatpally, Hyderabad, AP, India
³Infosys Technologies, Gachibowli, Hyderabad, AP, India

Abstract: The aim of this study is to develop a novel ontology ranking technique that ranks a given set of ontologies in a certain domain area. We examine each ontology by considering the OWL language constructs which are used to build that particular ontology. We define ontoweight measure and develop a methodology to compute the score and rank the ontologies. The main contributions of this study are adapting OWL constructs to determine the expressiveness of a given ontology, introducing ranking metric and methodology for ranking ontologies.

Key words: Ontology analysis, ontology ranking, OWL construct

INTRODUCTION

The semantic web is gaining importance due to its ability to interpret and process data and develop knowledge models which are useful for efficient search and retrieval. The semantic web is an extension of the current World Wide Web, in which information is given a well defined meaning enabling the computers and people to work with better cooperation. The semantic web provides a framework for sharing, combining and reuse of data across multiple applications with different requirements. One of the goals of the semantic web is to incorporate semantics so as to describe a concept such as travel, health, tourism, entertainment. Ontologies provide a flexible way of introducing semantics into the semantic web. They allow users to define their own vocabulary based on the existing ones. The major advantage of using ontologies is that they have the potential for the reuse of knowledge. An ontology developed by one person can be modified, extended or pruned by others as required, thereby avoiding the huge effort of starting from the scratch. A number of ontology libraries currently exist. Example libraries are Ontolingua (www.ksl.stanford.edu/software/ontolingua) and the OWL library (<http://protege.stanford.edu/plugins/owl/owl-library>). In order to get the right information, the search engines must be capable of finding the ontologies we are looking for. Some ontology search engines have already been developed, for instance, Swoogle developed by Ding *et al.* (2004) and Ontosearch developed by Zhang *et al.* (2005).

Current trends in knowledge management require working with multiple ontologies in order to support complex applications in a domain area. Web ontology language (OWL) is the approved standard which is used for developing ontologies in the semantic web. Due to

increase in multi-disciplinary applications, we will see a large growth of OWL ontologies. In this study, there arises a need to use more than one ontology to get the right kind of information the user is looking for. Finding a suitable ontology that serves the user's purpose becomes a hard task. In this study, we present a novel ontology ranking technique that ranks a given set of ontologies in a certain domain area. We examine each ontology by considering the OWL language constructs which are used to build that particular ontology. The main contributions of the study are adapting OWL constructs to determine the expressiveness of a given ontology, introducing ranking metric, methodology for ranking ontologies.

BACKGROUND

An ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them.

The main reasons for developing an ontology are:

- To share common understanding of the structure of information among people or software agents
- To enable reuse of domain knowledge
- To make domain assumptions explicit
- To separate domain knowledge from the operational knowledge
- To analyze domain knowledge

Web ontology language (OWL) is used for expressing the ontologies on the semantic web. The OWL is developed with an intention of providing a language that can be used to describe the classes and relations between them that are inherent in web documents. The

OWL facilitates greater machine interpretability of web content than that supported by XML, RDF and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics. The OWL has three increasingly-expressive sublanguages: OWL Lite, OWL DL and OWL Full.

The list of OWL Lite language constructs (www.w3.org/TR/2004/REC-owl-features/20040210) is given in Fig. 1. The OWL DL and OWL Full language constructs that are in addition to those of OWL Lite are given in Fig. 2. An ontology that describes a certain concept is developed based on these OWL language constructs. Ontology developers adopting OWL should consider which sublanguage best suits their needs. The choice between OWL Lite and OWL DL depends on the extent to which users require the more-expressive constructs provided by OWL DL. The OWL provides the following functionalities:

- Definition of OWL classes, using the owl:class construct, for the representation of sets of individuals sharing some properties. Class hierarchies may be defined using the rdfs:subclassOf construct

- Definition of OWL properties, for the representation of the features of the OWL class individuals. Two kinds of properties are provided by OWL:
 - Object Properties, defined using the owl:objectProperty construct, which relates individuals of one OWL class (the property domain) with individuals of another OWL class (the property range)
 - Datatype Properties, defined using the owl:datatypeProperty construct, which relates individuals belonging to one OWL class (the property domain) with values of a given datatype (the property range). Property hierarchies may be defined using the rdfs:subPropertyOf construct
- Definition of restrictions, using the owl:Restriction construct, which includes type restrictions, cardinality restrictions and value restrictions

OWL is used as a standard to express relationships between classes and properties in a given concept.

RDF Schema Features: Class(Thing, Nothing) rdfs:subClassOf rdf:Property rdfs:subPropertyOf rdfs:domain rdfs:range Individual	(In)Equality: equivalentClass equivalentProperty sameAs differentFrom AllDifferent distinctMembers	Property Characteristics: ObjectProperty DatatypeProperty inverseOf TransitiveProperty SymmetricProperty FunctionalProperty InverseFunctionalProperty
Property Restrictions: Restriction onProperty allValuesFrom	Restricted Cardinality: minCardinality(only 0 or 1) maxCardinality cardinality	Header Information: Ontology imports
Class Intersection: intersectionOf	Versioning: versionInfo priorVersion backwardCompatibleWith incompatibleWith DeprecatedClass DeprecatedProperty	Annotation Properties: rdfs:label rdfs:comment rdfs:seeAlso rdfs:isDefinedBy AnnotationProperty OntologyProperty

Fig. 1: OWL lite constructs

Class Axioms: oneOf disjointWith equivalentClass rdfs:subClassOf	Boolean Combinations of Class Expressions: unionOf complementOf intersectionOf
Arbitrary Cardinality: minCardinality maxCardinality cardinality	Filler Information: hasValue

Fig. 2: OWL DL and OWL full constructs

MOTIVATION

Here, we explain our approach by considering a sample camera ontology. As shown in Fig. 3, we have three sets of data. The first set is the keyword collection set that contains the user input keywords plus the words that are related to the concept camera, which are taken from wordnet (www.wordnet-online.com). The second set shows all the available OWL language constructs. The third set shows a sample camera ontology. Arrows are drawn to show how each OWL construct is used to describe the concept, camera. If an ontology contains a hundred percent match of class labels for all the keywords that are present in the key word collection set, then that ontology can be considered as closer to an ideal ontology. So, this way it is important to examine an ontology and see how many relevant class labels are matching with the keyword collection set. It is not sufficient if the class label alone matches with the keyword. We need to see how well the concept is described in terms of OWL constructs in that particular relevant class. Also, a concept can be described as a whole or part of an ontology. We need to see how much portion of the given ontology has the relevant OWL classes that describe the concept we are looking for.

Present study concentrates on two aspects (1) How well the concept is described in terms of OWL constructs

in a particular relevant class? (2) How much portion of the given ontology has the relevant OWL classes that describe the concept we are looking for? By measuring these two aspects we can arrive at a score using which the ontologies can be ranked. We call that score as ontoweight.

In this study, we considered OWL ontologies that describe the concept drug and tried to rank them based on the above mentioned OWL language constructs.

Several techniques to rank and choose the right ontology have been proposed. Jones and Alani (2006) proposed content based ontology ranking. The ranking is done according to how many of the concept labels in the set of ontologies match the set of keywords. An ontology which has more class labels that match the key words is deemed more suitable and is ranked higher than others. Alani *et al.* (2006) proposed AKTiveRank technique for ranking the ontologies. The AKTiveRank technique applies a number of analytic methods to rate each ontology based on an estimation of how well it represents the given search terms.

Yu *et al.* (2006) proposed ARRO (Approach for Ranking and Retrieving Ontologies). In this approach, the hierarchy of the ontology is regarded as one of the most important measures for ranking ontologies. The semantic relations among the classes of the ontologies will be measured and the logic views of the query terms are

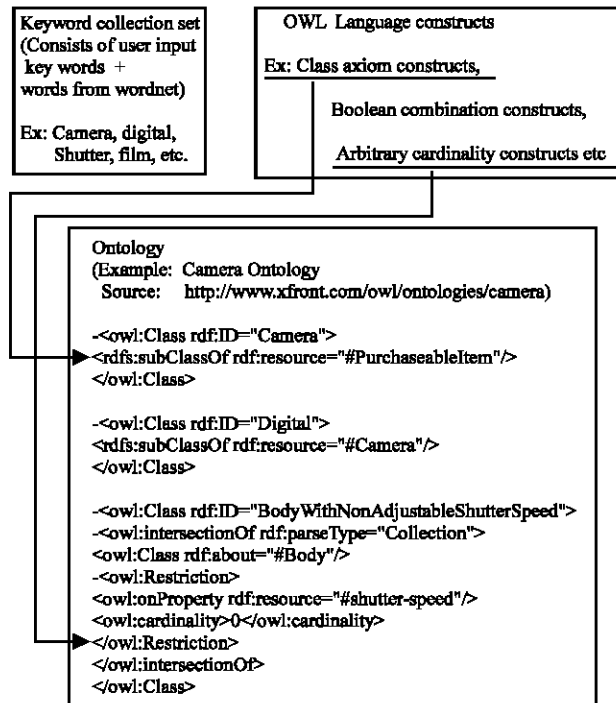


Fig. 3: The camera concept-described in terms of OWL constructs

formed. The ARRO can combine the logic inference and ranking together through the logic view.

Ding *et al.* (2004) proposed Swoogle, a search engine that a user can query for ontologies that contain specified key words which appear as class or property names. Patel *et al.* (2003) proposed Ontokhoj, a semantic web portal that can be used for ontology searching and ranking. Both Swoogle and Ontokhoj rank ontologies by using a PageRank like method developed by Page *et al.* (1999) that analysis the links and referrals between ontologies for identifying the most popular ontologies. However, the majority of ontologies on the web are poorly connected and more than half of them are not referred to by any other ontologies at all.

Ontosearch 2 is a search engine developed at the University of Aberdeen which can be used for querying and ranking ontologies. The public beta version is currently available at www.ontosearch.org. The current version is under revision and the next release of Ontosearch 2 will contain a keyword based search mechanism to allow broad searches of the entire repository of ontologies.

SemSearch is a search engine developed by Lei *et al.* (2006). It ranks ontologies according to their closeness to the specified user key words. It considers two factors-one is the matching distance between each key word and its semantic matches and the other is the number of key words the search results satisfy.

Samir and Arpinar (2007) developed OntoQA, a tool evaluates and ranks the ontologies based on two metrics: (1) Schema metrics which address the design of the ontology schema. The two metrics are relationship diversity and schema deepness. (2) Instance metrics: These are divided into three sub dimensions-Overall KB (Knowledgebase) metrics that evaluate the overall placement of instances with regard to the schema, class specific metrics that evaluate the instances of a specific class and compare it to the instances of other classes and relationship specific metrics that evaluate the instances of a specific relationship and compare it to the instances of other relationships.

Buitelaar *et al.* (2004) developed OntoSelect which is a dynamic web based ontology library. OntoSelect allows searching as well as browsing of ontologies according to size (number of classes, properties) representation format (DAML, RDFS, OWL), connectedness (score over the number of included and referring ontologies) and human languages used for class and object property labels.

Fernandez *et al.* (2006) proposed Collaborative Ontology Reuse and Evaluation (CORE)-a tool that receives an informal description of a semantic domain and determines which ontologies are the most appropriate to describe the given domain. The ranking of ontologies is done by using the rank fusion techniques.

Li *et al.* (2007) proposed OntoLook-a relation based search engine. OntoLook records all of the relations among keywords and concepts and sends these pairs to the ontology database. It then searches the ontology database and retrieves ontologies which satisfy most of the keyword-concept pairs. However, it does not rank the ontologies.

Based on the various evaluation and ranking methods mentioned above, it is clear that there is a need to assess all important features of an ontology and rank the ontologies from a given set of ontologies. This study concentrates on examining the ontologies based on OWL language constructs and ranks them.

PROPOSED WORK

The proposed ontology ranking architecture is given in Fig. 4.

The main component of our ranking architecture is the ontology ranking engine. The keywords are sent to the engine. These keywords are sent to the wordnet database and all the concept related words are collected from the wordnet. Then the ranking engine calculates ontoweight by considering all the OWL constructs that are present in the ontology.

The algorithm for the ranking technique is given in Fig. 5.

The explanation of the ranking technique is given in Fig. 6.

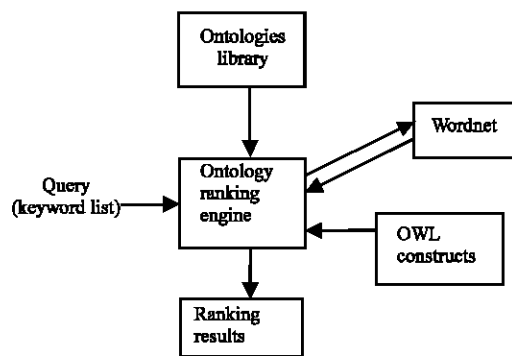


Fig. 4: Proposed ontology ranking architecture

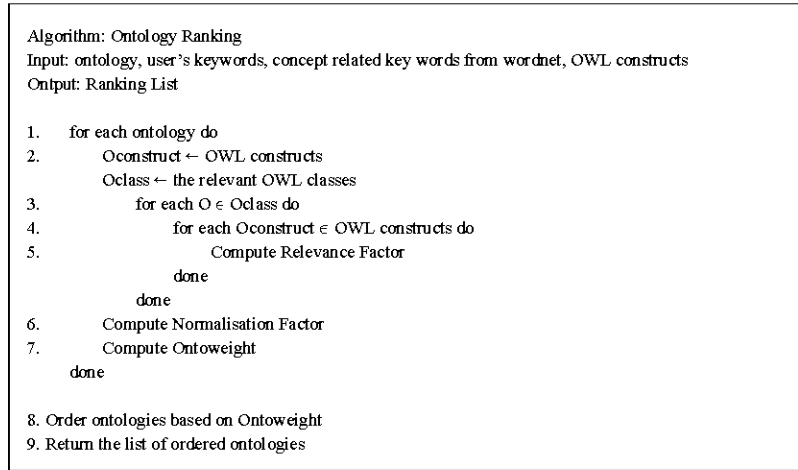


Fig. 5: Algorithm for the proposed ranking technique

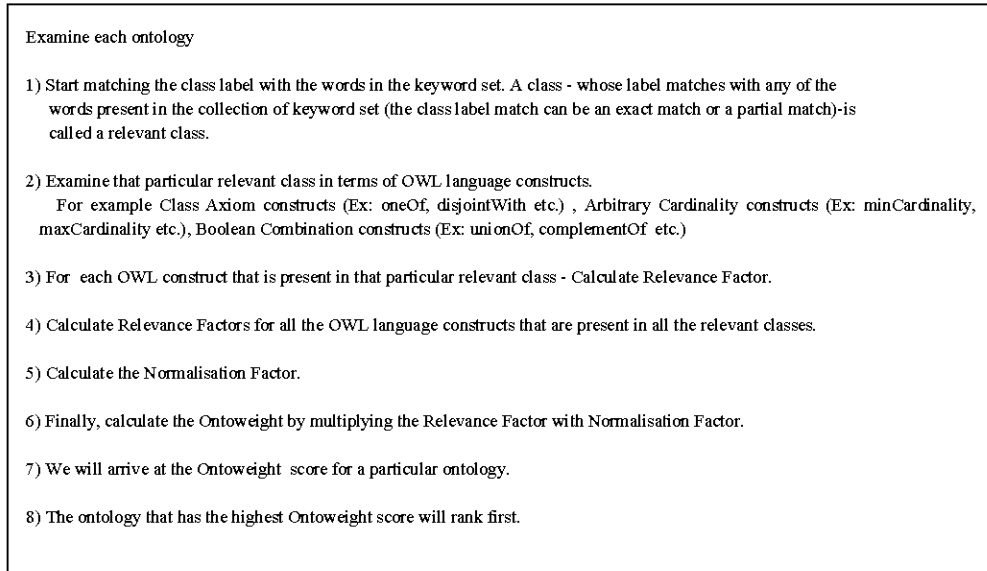


Fig. 6: Explanation of the proposed ranking technique

THE RANKING PROCEDURE

An ontology can be treated as a collection of owlClasses.

Let O denote an ontology, C denote an individual owlClass in that ontology.

The ontology can be represented as follows:

$$O = (C_1, C_2, C_3, C_4, \dots, C_n)$$

In the above set of classes, we consider only the classes whose labels match with the key words in the keyword collection set. These classes are the relevant OWL classes and we will consider each of these classes and examine them for OWL constructs.

All the constructs that are available in the web ontology Language can be represented as a collection.

Let OC denote an individual OWL Construct.

$$\text{OWL constructs} = (OC_1, OC_2, OC_3, OC_4, \dots, OC_n)$$

We look for the occurrences of each of the above constructs in all of the relevant OWL Classes.

Salton and Buckley (1988) presented several term-weighting approaches in automatic text analysis. The main function of a term-weighting approach is the enhancement of retrieval effectiveness. Our work is based on their study. For each OWL construct-we look for its occurrence in a relevant OWL class and calculate its

weightage. While calculating the weightage, we also consider the number of times the same OWL construct occurs in all of the relevant OWL classes throughout the ontology document. The final weightage that we get for an OWL construct is called as relevance factor for that particular OWL construct.

The Relevance Factor (RF) for each OWL construct in each individual relevant OWL class is calculated as follows:

Let RF_{oc} denote Relevance Factor for an OWL construct.

$$RF_{oc} = \alpha \times f_{oc,c} \times (1/\log(N/f_{oc,N})) \quad (1)$$

Where:

$f_{oc,c}$ = The number of times the OWL construct occurs in that particular class

N = The total number of relevant OWL classes in the entire ontology and

$f_{oc,N}$ = The number of relevant classes in which the OWL construct (OC) appears

α takes the value of either 1 or 0.4. If the class label exactly matches with the key word then the value is taken as 1. If the class label matches partially with the keyword then the value is taken as 0.4.

The Eq. 1 gives the relevance factor for only one OWL construct in only one relevant class.

Let us say there are N number of relevant OWL classes in the given ontology.

The Eq. 1 can be written as follows:

$$RF_{oc} = \sum_{j=1}^N (\alpha \times f_{oc,c_j} \times (1/\log(N/f_{oc,N}))) \quad (2)$$

where, f_{oc,c_j} is the number of times the OWL construct occurs in the class j .

The Eq. 2 gives the Relevance Factor of only one construct in all the relevant OWL classes. Since, an ontology can be expressed in any of the OWL constructs, we need to calculate the Relevance Factor for all the constructs that are available in the OWL.

Therefore, the Eq. 2 is modified as follows:

Assume that the total number of constructs available in OWL is k .

$$RF = \sum_{i=1}^k \sum_{j=1}^N (\alpha \times f_{oc_i,c_j} \times (1/\log(N/f_{oc_i,N}))) \quad (3)$$

Where:

f_{oc_i,c_j} = The number of times the OWL construct, i occurs in the class j

$f_{oc_i,N}$ = The number of relevant classes in which the OWL construct, i appears

Using the Eq. 3, we can examine each relevant class and come up with relevance factor for all the constructs in a given ontology.

Calculation of Normalisation Factor (NF): The NF takes care of the length of the ontology. For example, if we are looking for the concept brake we may find an ontology exclusively describing the brake. We may also find the same concept described in an automobile ontology. However, in the automobile ontology, the concept brake forms just a part of it.

So, a concept can be described as a whole or part of an ontology. The NF gives us an indication of how much portion of the given ontology has the relevant OWL classes that describe the concept we are looking for.

$$NF = \frac{\text{No. of relevant OWL classes in the ontology}}{\text{Total No. of OWL classes present in the ontology}}$$

The ontoweight is calculated by multiplying the Relevance Factor (RF) with the Normalisation Factor (NF).

$$\text{Ontoweight} = RF \times NF$$

The ontology with the highest ontoweight is ranked first.

EXPERIMENT

We considered the drug as a sample concept and downloaded a set of 10 ontologies on this concept by using the semantic web search engine, Swoogle. We conducted the experiment and calculated the ontoweight for each ontology. The list of ontologies that are considered are shown in Table 1. The results are presented in Table 2.

Adding concept related terms to the key word set: The terms that are related to the concept drug are taken from the wordnet. These terms are added to the terms in our keyword set. Now, our keyword set consists of the concept word drug plus some more terms that are related to the concept. The addition of the terms that are related to the concept will give us a better keyword set and by using all the terms in the thus formed keyword set, we can examine each ontology in a more meaningful manner.

The list of the terms for the concept drug that are obtained from the wordnet are included in Table 2.

Table 1: Test set of ontologies (Concept: drug)

Ontology	Ontology URL
Drugont6	http://www.phinformatics.org/Assets/Ontology/umls.owl
Drugont8	http://www.bio.cs.titech.ac.jp/~dio/dio_event.owl
Drugont10	http://spire.umbc.edu/ontologies/EthanKeywords.owl
Drugont20	http://www.mindswap.org/2004/multipleOnt/FactoredOntologies/NCI/Ontology13.owl
Drugont23	http://www.mindswap.org/2004/multipleOnt/FactoredOntologies/NCI/Ontology17.owl
Drugont24	http://www.mindswap.org/2004/multipleOnt/FactoredOntologies/NCI/Ontology1.owl
Drugont27	http://spire.umbc.edu/ontologies/taxa_2008_04_08/old_style/EthanConcepts.owl
Drugont35	http://ontology.dumontierlab.com/pharmacogenomics-complex-2.1.owl
Drugont40	http://www.w3.org/2005/04/swls/ls-ont-1.0
Drugont55	http://proton.semanticweb.org/2006/05/protonu

Table 2: List of terms for the concept drug

Medicine	Anesthetic	Diuretic	Levallorphan
Narcotic	Anaesthetic	Water pill	Lorfan
Agent	Antagonist	Abuse	Medication
Pharmacopoeia	Antisiphilitic	Street drug	Medicament
Abortifacient	Anti-trif compound	Feosol	Medicinal
Aborticide	Brand name	Fergon	Mydriatic
Abortion inducing	Proprietary	Fertility drug	Pentoxifylline
Agonist	Controlled substance	Generic drug	Trental
Anesthetic	Dilator	Intoxicant	Psychoactive
Mind altering	Consciousness altering	Psychotropic agent	Relaxant
Soporific	Hypnotic	Stimulant	Suppressant
Appetite	Synergist	Virility drug	Anti-impotence
Medicate			

Table 3: Ontologies ranked as per ontoweight

Ontology ID	Ontoweight	Rank
Drgont8	450.519	1
Drgont35	145.727	2
Drgont20	59.653	3
Drgont23	0.244	4
Drgont6	0.176	5
Drgont40	0.088	6
Drgont24	0.074	7
Drgont55	0.011	8
Drgont27	0.003	9
Drgont10	0.002	10

RESULTS AND DISCUSSION

The ranking of ontologies as per the ontoweight is shown in Table 3.

By doing a manual inspection of the above ontologies, it can be verified that the results we got are correct. It is very difficult to pinpoint the right selection of parameters or structural properties to investigate while ranking a set of ontologies. The ranking can be dependent on personal preference as well as the purpose for which the ontology is used. The ontoweight metric can be used to see how well an ontology represents a certain concept in terms of OWL language constructs. This method also takes care of how much portion of the given ontology has the relevant OWL classes that describe the concept we are looking for. Based on these two aspects we tried to rank the ontologies.

CONCLUSION

Ranking ontologies is an important component of a semantic web search engine. As the semantic web

emerges, a large number of ontologies will be developed. Finding the relevant ontology that satisfies all the user's requirements becomes a challenging task. In this study, we proposed a new ranking technique that ranks ontologies based on OWL language constructs. Combining this technique with other ranking techniques would be beneficial in the ontology ranking process.

REFERENCES

- Alani, H., C. Brewster and N. Shadbolt, 2006. Ranking ontologies with AKTiveRank. Proceeding of the 5th International Semantic Web Conference (ISWC), Aug. 29-Jan. 6, Georgia, USA., pp: 1-15.
- Buitelaar, P., T. Eignar and T. Declercq, 2004. Onto Select: A dynamic ontology library with support for ontology selection demo session. Proceeding of the Demo Session at the International Semantic Web Conference, Aug. 17, Hiroshima, Japan, pp: 1030-1033.
- Fernandez, M., I. Cantador and P. Castells, 2006. CORE: A tool for collaborative ontology reuse and evaluation. Proceedings of the 4th International Workshop on Evaluation of Ontologies for the Web, at the 15th International World Wide Web Conference, CEUR Workshop Proceedings, Vol. 179, May 2006, Edinburgh, UK., pp: 1-8.
- Jones, M. and H. Alani, 2006. Content-based ontology ranking. Proceedings of the 9th International Protégé Conference, July 23-26, Stanford, California, USA., pp: 1-4.

- Ding, L., T. Finin, A. Joshi, R. Pan and R.S. Cost *et al.*, 2004. Swoogle: A semantic web search and metadata engine. Proceeding of the 13th ACM Conference on Information and Knowledge Management, Nov. 09, Department of Computer Science and Electronic Engineering, USA., pp: 652-659.
- Lei, Y., V. Uren and E. Motta, 2006. SemSearch: A search engine for the semantic web. Proceeding of the 15th International Conference on Managing Knowledge in the World of Network (EKAW), Feb. 14, Podebrady, Czech Republic, pp: 238-245.
- Li, Y., Y. Wang and X. Huang, 2007. A relation-based search engine in semantic web. IEEE Trans. Knowledge Data Eng., 19: 273-281.
- Page, L., S. Brin, R. Motwami and T. Winograd, 1999. The pagerank citation ranking: Bringing order to the web. Technical Report Stanford InfoLab. <http://ilpubs.stanford.edu:8090/422/>.
- Patel, C., K. Supekar, Y. Lee and E. Park, 2003. Ontokhoj: A semantic web portal for ontology searching, ranking and classification. Proceeding of the 5th ACM International Workshop on Web Information and Data Management, Nov. 7-8, New Orleans, Louisiana, USA., pp: 58-61.
- Salton, G. and C. Buckley, 1988. Term-weighting approaches in automatic text retrieval. Inform. Process. Manage., 24: 513-523.
- Samir, T. and B. Arpinar, 2007. Ontology evaluation and ranking using OntoQA. Proceedings of the 1st International Conference on Semantic Computing, Sept. 17-19, Irvine, CA, pp: 185-192.
- Yu, W., J. Cao and J. Chen, 2006. A novel approach for ranking ontologies on the semantic web. Proceeding of the 1st International Symposium on Pervasive Computing and Applications, Aug. 3-5, Urumqi, pp: 608-612.
- Zhang, Y., W. Vasconcelos and D. Sleeman, 2005. Ontosearch: An ontology search engine. Proceeding of the 24th SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, Dec. 24, Springer, London, pp: 58-69.