Integration of Online Ontologies Using Combined Ranking Algorithm

C.K.Sarumathy, J.Gokulraj, P.Selvaperumal

Abstract—

In the Semantic Web, knowledge representation is largely based on ontologies. Ontology should be constructed in a way such that it should meet the requirements of the users. The main difficulty involved in the construction of ontologies is the high cost incurred in building them. Gathering complete knowledge about a specific domain requires more time and it doesn't guarantee that the resulting ontology will be better than the existing ontologies. Hence, an approach for reusing the existing ontologies to build new ontologies has been proposed. This process makes use of the following steps: identification of existing ontologies, use of combined ranking algorithm (OntoRank+AKTive),segmentation and integration. As a result, best quality ontology can be obtained.

Keywords

AKTive Rank, Extraction, Fragmentation, Onto Rank, Reusability.

1. Introduction

Ontology is an explicit specification of conceptualization – Tom Gruber. As far as ontologies are concerned, they are intricate to construct[2]. This construction process is also a cumbersome task to understand for the users. Research works are going on for minimizing the cost incurred in the construction of ontologies. Many approaches have been invented for extracting ontologies from existing knowledge bases. After the extraction, those ontologies which have the rich representation of domain knowledge should be chosen by ranking them. As a result of this phase, the top ranked ontologies will be obtained. These top ranked ontologies may contain This paper focuses on the steps for reusing the existing ontologies by ranking the best among them. No special tools have been constructed so far that enables the easy construction of new ontologies from the existing ones.

2. Why reuse and ranking of ontologies is important?

One of the problem encountered in the ontology construction task is sometimes a newly constructed ontology may result in an ontology which looks almost similar to the already existing ontology making the work useless at last. So, the reusing of existing ontologies will serve as a boon to this problem.[1]

Ontology ranking plays a vital role in the integration of existing ontologies. The extraction of existing ontologies from the sources is not a very difficult task. A semantic web search engine namely Swoogle[5] is available which provides the ontologies for the user. But analyzing all those ontologies and choosing the best one that gives the better representation of concepts about a particular domain is a back-breaking work as some ontologies are larger in size. In addition to this, analysing ontologies is a time consuming task. So, to raise over this difficulty, an efficient ranking system needs to be employed. This ranking system should be capable of analyzing all the available ontologies for a domain based on a certain criteria. Generally, the ontologies are ranked based on various criteria such as page citation count, frequency of occurrence of the keyword given in the query and so on.

Swoogle contains more than 10,000 ontologies and it employs the page ranking algorithm. But it doesn't have any factors to do with the user's query. This will result in the hiding of few links which will contain valuable concept representations. Hence, the need for ranking system arises to provide the user with the best set of links.

After ranking the ontologies, segmentation is done. The integration of different fragments segmented from several top ranked ontologies is carried out. As a result of ranking phase, the user is provided with the best set of ontologies. Each ontology contains several fragments that contain different concept representations. These dissimilar concepts are identified and finally these segments are integrated to give the better domain representation.

3. Existing System

Content based ranking algorithm

Content based ranking algorithm[3], developed by the University of Southampton ranks the ontology by analyzing how well it covers a domain of interest. According to this algorithm, an ontology which contains more class labels that match with the keyword will be given the highest rank. This algorithm makes use of the WordNet thesaurus. It helps to disambiguate the terms which contain more than one meaning. Here, the Class Match Score is calculated by the formula,

$$\text{CMS}\,[o \in O] = \sum_{i=1}^{n} I(Pi, O) * 5\log(n + 2 - i)$$

The parameter I(Pi,O) is defined as,

$$\begin{cases} 1 : & \text{if } o \text{ contains a class with label matching } Pi \\ 0.4 : & \text{if } o \text{ contains class with label containing } Pi \\ 0 : & \text{if } Pi \text{ does not appear in any of class labels} \end{cases}$$

where O is the set of ontologies, P represents the set of potential class labels, n is the number of terms collected from the corpus and i represents the number of search terms in a given

Manuscript received 01 September, 2011.
C.K.Sarumathy P.G Scholar,Dept of CSE,Sona college of technology salem,Tamil Nadu.
J.**Gokulraj**,Assistant professor Department of IT, V.S.B Engineering college, Karur,Tamil nadu.
P.Selvaperumal Assistant professor V.S.B Engineering college, Karur.Tamil Nadu.

query. The values 1 and 0.4 are allotted for exact and partial matches respectively. The drawback of this algorithm is it doesn't take any attempt to look at how well the concepts are connected.

OntoRank algorithm

OntoRank algorithm [15][16] employs the link analyze method. Here the ontology is ordered by the evaluation of their importance. This reflects the influence of one ontology exerted to another. Two concepts C1 and C2 can be used as a reference relationship if and only if a relation exists between the two classes in the relationship set {rdf:type, rdfs:subclass, rdfs:domain, rdfs:range}. This algorithm evaluates the ontology importance statically, but it doesn't take user's query as an affect factor.

AKTive rank algorithm

The AKTive rank algorithm,[4] calculates the matching degrees between ontology and multiple keywords given by user. Hence, it returns more accurate results to the user. The algorithm uses four kinds of evaluation measures as follows,

I) Class Match Measure (CMM),
11) Centrality Measure (CEM), III) Density Measure (DEM),
IV) Semantic Similarity Measure (SSM). [20]

Based on the weighted summation of these four kinds of measures, the ontology rank score is calculated. The disadvantage of this algorithm is it takes more time to respond to user's query. Even though many ranking algorithms are in existence, they are not effective. Hence, in the proposed system a combined ranking algorithm (OntoRank + AKTive rank) is employed to overcome the drawbacks of existing algorithms.

4. Proposed System

Ontology Integration Process Using Combined Ranking Algorithm

The ontology integration from existing ontology resources requires the following semantic web technologies such as

i) Extraction of ontologies from Swoogle,
ii) Combined ranking algorithm,
iii) Extraction of fragments and
iv) Integration of fragments

4.1. Extraction of ontologies

The first step deals with the extraction of existing ontologies from web sources by using keyword-based search method in Swoogle. This process results in the retrieval of large set of ontology URI's. Here, the process of selection of the URI's is done by just analyzing the links.

4.2. Ranking algorithm

4.2.1 Combined Ranking Algorithm

The output of the first phase provides the user with a set of links. An apt algorithm should be applied to find and rank out the ontologies with the rich set of concept representations. To rank, a combination of OntoRank and AKTive algorithm is to be used and the algorithm comprises of the following steps,

1) Extend the concept of the Query

Assume that D is the ontology set in the semantic web, then the ontology set D = {d1, d2,d3,…dn}, dj ε D (1≤ j ≤ n) ; x_i denotes a set composed of all concept a_i which has the similar semantic relation with the semantic concept x_i. If x,y,z represent different concepts within d_j then this will be defined as the logical view LV1,LV2,LV3 respectively. These three can be described by logical view LV in the semantic concept x_i. In this way the concept x, y and z all will belong to the concept set $[X_i]$.

The logical view LV (Logical View) of concept X is calculated as follows:

$LV(X) = \alpha * extension(X) + \beta * include(X) + X$ where extension(X) is the connotation of concept X, and include(X) is the extension of concept X.

Here, α and β the percentage of concept X connotation and extension respectively. In order to extend the scope of synonyms concept dictionary is used.

According to TF/IDF (Term Frequency-Inverse Document Frequency), the weight w([Xi],d) of semantic concept [Xi] can be calculated by,

$$w([Xi],d)=log(f([Xi],d)+1)*log(n/n_i+1)$$

where w([Xi],d) is the weight of semantic concept set to ontology d; f([Xi],d) denotes the occurrence frequency of the element xi; n denotes the total numbers of ontology in the set D; n_i denotes the number of ontologies that contain the semantic concept x_i.

2) Calculation of importance of ontology

Here, the OntoRank algorithm is used to calculate the accessed probability of ontology ' a'.

$$Ontorank(\alpha) = wPR(\alpha) + \sum_{x \in orc(\alpha)}^{n} wPR(x)$$

wPR(α) is the accessed probability of SWD 'a' (Semantic Web Document) of itself and denotes the accessed probability of all imported SWD.

3) Application of AKTive Rank algorithm

After applying the OntoRank ranking algorithm, the AKTive ranking algorithm[4] is applied to calculate the score of the ontologies. This algorithm gives the score by manipulating the relevant terms of the query. The score is calculated by,

$$Score[o \in O] = \sum_{i=1}^{4} \frac{W_i \quad M[i]}{\max_{1 \le j \le |o|} M[j]}$$

where M indicates four kinds of measures ,
M = {M[1], M[2], M[3], M[4]} = {CMM, CEM, DEM, SSM }, O denotes the ontology and w_i is the weight factor and it's normalized to have the values between zero and one.

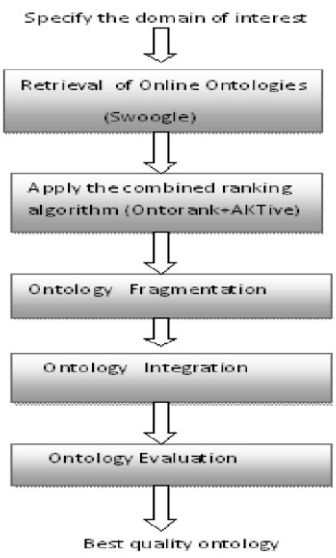

Figure 1: System Architecture

4.3) Fragmentation of ontology

The ranking phase will return the ontologies that contain rich concept representations and it should be noted that all the contents of the ontology are not worth Therefore, the concepts that add value to a particular domain are selected and the remnants are discarded. Sometimes the size of the additional representations[19] will be very large. In order to avoid such oversize representations, depth limit is set. A methodology for segmenting the ontologies based on property filtering and bound depth limiting is adopted for this purpose [7]. Apart from this, some simplistic and complex approaches are available for segmenting the ontologies[8][9]. Based on the perspective, ontologies are also segmented based on the queries of an application [10].

4.4) Integration of fragments

The fragmentation phase returns the set of fragments that are segmented from different owl files[18]. These fragments are merged based on the Ontology Merging algorithm[21]. The major function of the algorithm is a similarity search function that takes the concept and looks for its more similar concept in the ontology, giving back the most similar concept and a sv *(*similar value) with value between 0 and 1. A number of tools such as PROMPT Suite [12] which is integrated into the Prot´eg´e ontology editor [6][11] assist in the mapping and merging of ontologies. The output of integration phase is the ontology of best quality.

4.5) Ontology Evaluation

The evaluation is done to check whether the system meets the minimum quality constraints. However, the evaluation helps to identify the semantic gaps and other inefficiencies, if present. As it is known that ontology construction is a cyclic process. Hence, modifications are done to improve the quality of the ontology in future. Even though some approaches like tagging[14] and Onto Clean[13] are in existence they prove to be very costly. So, the manual evaluation involving the experts opinion is done.

5) Conclusion and Future Work

Ontology reusability process helps to save the development time of the researchers. Moreover, the combined ranking algorithm enables the user to find the desired ontology quickly. This algorithm grants the high quality ontology. Many of the ontologies constructed by developers are not put on the web. The idea of reusing the existing ontologies will help to tackle over this problem.

This idea is made better by allowing the user's of the system to modify, delete from and add to the automatically built ontology. It is also possible to do improvements in the ranking phase, by modifying the parameters of the algorithm.

References

[1] An Ontology Approach to creating a New Recipe by Reusing Parts of the Existing Recipe,IEEE,2009,Rie Makino, Ichiro Kobayashi, Noriaki Izumi and Koiti Hasida.

[2] Harith Alani. Position Paper: Ontology construction from online ontologies. In Proc. 5th International World Wide Web Conference, Edinburg, 2006

[3] Content-based Ontology Ranking, Matthew Jones, Harith Alani

[4] Ranking ontologies with AKTive Rank, Harith Alani, Christopher Brewster and Nigel Shadbolt.

[5] http://swoogle.umbc.edu

[6] http://wordnet.priceton.edu

[7] J. Seidenberg and A. Rector. Techniques for segmenting large description logic ontologies. In *Workshop on Ontology Management: Searching, Selection, Ranking, and Segmentation. 3rd Int. Conf. Knowledge Capture (K-Cap)*, pages 49–56, Banff, Canada, 2005.

[8] N. F. Noy and M. A. Musen. Specifying ontology views by traversal. In *3rd Int. Semantic Web Conf. (ISWC'04)*, Hiroshima, Japan, 2004. [9] M. Bhatt, C. Wouters, A. Flahive, W. Rahayu, and D. Taniar. Semantic completeness in sub-ontology extraction using distributed methods. In *Proc. Int. Conf. on Computational Science and its Applications (ICCSA)*, pages 508–517, Perugia, Italy, 2004. LNCS, Springer Verlag.

[10] H. Alani, S. Harris, and B. O'Neil. Ontology winnowing: A case study on the akt reference ontology. In *Proc. Int.Conf. on Intelligent Agents, Web Technology and Internet Commerce*

[11] M. A. Musen, R. W. Fergerson, W. E. Grosso, N. F. Noy, M. Y. Grubezy, and J. H. Gennari. Component-based support for building knowledge-acquisition systems. In *Proc. Intelligent Information Processing (IIP 2000) Conference of the International Federation for Processing (IFIP), World Computer Congress (WCC'2000)*, pages 18– 22, Beijing, China, 2000.

[12] N. F. Noy and M. A. Musen. The prompt suite: Interactive tools for ontology merging and

mapping. *International Journal of Human-Computer Studies*, 59(6): 983–1024, 2003.

[13] N.Guarino & C.Welty. Evaluating ontological decisions with Ontoclean. *Communications of the ACM*, 45(2):61–65, 2002.

[14]J. Volker, D. Vrandecic, and Y. Sure. Automatic evaluation of ontologies (aeon). In *Proc. 4th Int. Semantic Web Conf. (ISWC)*, Galway, Ireland, 2005.

[15] Ding L, et al Finding and Ranking Knowledge on the Semantic Web. In the Proceedings of the 4th International Semantic Web Conference.

[16] Ding L, et al. Swoogle: a search and metadata engine for the semantic web. In : Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management, 2004, 652-659.

[17] Alani H, Brewster C. Ontology ranking based on the analysis of concept structures. Proceedings of the 3rd International Conference on Knowledge Capture, 2005,51-58.

[18] OWL Web ontology Language Overview. http://www.w3.org/TR/owl-features/

[19] P.C.Weinstein and W.P.Birmingham. Comparing concepts in differentiated ontologies. In Proceedings of 12th Workshop on Knowledge Acquisition.

[20] A.Maedche and S.Staab. Measuring similarity between ontologies. In Proc. European Conf. On Knowledge Acquisition and Management (EKAW), pages 251-263 , Madrid, 2002.

[21] A Language and Algorithm for Automatic merging of ontologies,

Alma Delia Cuevas Rasgado and Adolfo Guzman Arenas, In Proceedings of the 15th International Conference on Computing (CIC'06) , 2006. Modeling and Management (KAW' 99), Banff, Alberta, Canada, 1999.