Aslib Journal of Information Management

A novel ontology matching approach using key concepts

Tayybah Kiren Muhammad Shoaib

The current issue and full text archive of this journal is available on Emerald Insight at:
www.emeraldinsight.com/2050-3806.htm

A novel ontology matching approach using key concepts

Tayybah Kiren and Muhammad Shoaib
University of Engineering and Technology, Lahore, Pakistan

Received 10 April 2015
Revised 28 October 2015
Accepted 2 November 2015

Abstract

Purpose – Ontologies are used to formally describe the concepts within a domain in a machine-understandable way. Matching of heterogeneous ontologies is often essential for many applications like semantic annotation, query answering or ontology integration. Some ontologies may include a large number of entities which make the ontology matching process very complex in terms of the search space and execution time requirements. The purpose of this paper is to present a technique for finding degree of similarity between ontologies that trims down the search space by eliminating the ontology concepts that have less likelihood of being matched.

Design/methodology/approach – Algorithms are written for finding key concepts, concept matching and relationship matching. WordNet is used for solving synonym problems during the matching process. The technique is evaluated using the reference alignments between ontologies from ontology alignment evaluation initiative benchmark in terms of degree of similarity, Pearson's correlation coefficient and IR measures precision, recall and *F*-measure.

Findings – Positive correlation between the degree of similarity and degree of similarity (reference alignment) and computed values of precision, recall and *F*-measure showed that if only key concepts of ontologies are compared, a time and search space efficient ontology matching system can be developed.

Originality/value – On the basis of the present novel approach for ontology matching, it is concluded that using key concepts for ontology matching gives comparable results in reduced time and space.

Keywords Ontology matching, Ontology, Degree of similarity, Effectiveness measures, Key concepts, Ontology heterogeneity

Paper type Research paper

1. Introduction

The World Wide Web (WWW) is widely used as a worldwide medium for information sharing. Interoperability among disparate information systems in the WWW is limited because of the information heterogeneity (Mao, 2008). Ontologies are recommended as a way to resolve this problem by expressing and exchanging the information in a formal and explicit way. However, ontologies themselves suffer from the problem of heterogeneity. Ontology developers from different domains may interpret the same data in different ways and develop the ontologies having similar concepts represented with different names. This leads to heterogeneity, such as variation in naming and describing a concept in different levels of detail. This heterogeneity hampers the interoperability among distributed information systems. Ontology heterogeneity occurs, for instance, when one ontology only has the concept humans, while another ontology has further male and female sub-concepts, or when semantically similar classes in two ontologies have different instances. When terms in two ontologies have a different syntax but the same meaning, this is also called ontology heterogeneity. A clearer indication of the existence of ontology heterogeneity is experienced while querying through the SWOOGLE semantic web search engine. If we query for the concept "Apple," it returns 254 results for apple. In some of the resulting semantic web documents (RDF or owl) apple means a computer and some results represent apple as a fruit. Also the same concept can be

Aslib Journal of Information Management
Vol. 68 No. 1, 2016
pp. 99-111

2050-3806
DOI 10.1108/AJIM-04-2015-0054

described using different names in different ontologies, e.g. subject and book title have similar meanings in different ontologies. Thus, finding similarities between the two ontologies is a very important task.

Many techniques have been developed for ontology matching. Internal matching techniques use internal structure of the ontology to find similarities. External matching techniques use external resources, such as upper-level ontologies, synonym lists or dictionaries, as background knowledge for finding similarities between two ontologies. Existing ontology matching techniques, whether internal or external, compare all concepts of both ontologies to be matched. For large size ontologies, this task of comparing all the concepts to find similarity between the ontologies makes the ontology matching technique very slow and time consuming. The running time shows the efficiency characteristics of ontology matching tools. Efficient execution can be achieved by using a big amount of main memory, or efficient CPU. Therefore, usage of main memory should also be measured or improved. In conclusion, the challenge is to determine time and search space efficient ontology matching approaches (Shvaiko and Euzenat, 2013). A survey on ontology alignment techniques says that in the upcoming years, the field of ontology matching will realize a major thrust in discovering techniques that are capable of solving the heterogeneity issue regarding ontologies in an increasingly reduced amount of time (Marcos Martínez *et al.*, 2009).

The deep web consists of many online databases. These databases are accessed through their query interfaces. Each query interface takes queries over its query schemas. Efficient matching of query schemas is required in order to provide results to the user query in a reduced response time. For example, a user who wants to purchase a book from the web, may often need to search through alternative sources available for books (e.g. amazon.com and bn.com). Both of these sources refer to the same concepts with different names, i.e. at amazon.com there are concepts, author, title, ISBN, while in bn.com there are concepts, title of book and author's name. Thus a proposal of an ontology matching technique with reduced time and search space requirements is needed for supporting query mediation across deep web sources.

2. Related work

Using external sources to find similarities between the ontologies is one of the ten challenges to ontology matching (Shvaiko and Euzenat, 2008). Upper ontologies have been used as an external resource for finding similarities (Jain *et al.*, 2011; Li, 2004; Mascardi *et al.*, 2010). Use of upper ontologies for finding similarities restricts the solution to a particular domain. WordNet is used as an external resource for ontology matching in Agreement Maker (Cruz *et al.*, 2009), YAM++ (Ngo and Bellahsene, 2012) and LogMAP (Jiménez-Ruiz and Grau, 2011a, b). Wiktionary is used as background vocabulary source for ontology matching (Lin and Krizhanovsky, 2011). WikiMatch (Hertling and Paulheim, 2012) uses Wikipedia for ontology matching. It searches Wikipedia for finding the relevant documents for each concept of the input ontologies. Two concepts are considered to be similar if the number of common documents retrieved by these concepts is greater than a particular threshold. The problem with this approach is the execution time required to search the huge Wikipedia. Contextual information is also being used to help in matching ontologies. Google search engine is used as an external resource for ontology matching (Gligorov *et al.*, 2007). This approach is not suitable for large ontologies because it makes a large number of calls to a search engine. In Lin *et al.* (2010) context-based ontology matching is explored. Background ontology is used as context to find out similarities between the ontologies (Aleksovski *et al.*, 2006). In this technique the input ontologies have no structure at all. Online ontologies are used as background knowledge for ontology mapping by Sabou *et al.* (2008). It uses SWOOGLE to automatically search for the relevant semantic knowledge available online. Matching ontologies using instances is carried out by Breitman *et al.* (2008). This approach assumes that if two ontologies belong to similar domains, then their lexically same concepts are supposed to be matched. Alasoud *et al.* (2008) proposed a neighbor search algorithm and used this algorithm for similarity measure between ontologies. It took an average of 6,632.66 milliseconds for the conference track data set of ontology alignment evaluation initiative (OAEI). Along with quality and accuracy, the efficiency of ontology matching tools is of major significance in vibrant applications, particularly, when a user is not willing to wait very long for the system to respond or when the memory is limited. Current ontology matching techniques are mostly not optimized for resource utilization. Many recent systems have tried to tackle the efficiency matter, COMA++ (Do and Rahm, 2007) and Anchor-Flood (Seddiqui and Aono, 2009) applied segment-based approach to reduce space and time requirements. PORSCHE (Saleem *et al.*, 2008) and XClust (Lee *et al.*, 2002) proposed to increase the efficiency by reducing the need of clustering while integrating different types of matchers in their systems. Although these efforts were quite hopeful, they still lacked in achieving significant progress. For example, in OAEI-2007 (Euzenat *et al.*, 2007), only a few systems, such as Falcon (Hu *et al.*, 2008), took several minutes to finish the matching job, whereas other tools took much more time (hours and even days). In OAEI-2009 (Euzenat *et al.*, 2009), Anchor-Flood (Seddiqui and Aono, 2009) performed the job in 15 seconds.

Mork and Bernstein (2004) attempted to reduce time complexity by applying a simple and quick matcher. It tried to reduce the factor t in $O\ (n^2 \times t)$. But this system was unable to solve the time complexity issue because n^2 is a large number and the reducing factor t has an insignificant effect on the matching performance. Mao (2008) proposed to use parallel processing strategy to deal with the similarity calculation. The parallel processing idea is very simple and easy to be implemented; however it requires expensive hardware resources to set up the parallel processing environment.

Malasco (Paulheim, 2008; Do and Rahm, 2007) and Falcon-AO (Hu and Qu, 2008) proposed to apply a divide-and-conquer strategy in order to reduce the factor n^2 in $O\ (n^2 \times t)$. These techniques partition the ontology into blocks or modules to reduce the time complexity. The problem with these techniques is the size of the modules; the existing approaches for modularizing the ontology produce too large or too small modules. Moreover, after dividing ontologies into modules, concepts near borders are possibly to lose useful semantic information. As a result, the quality of ontology matching may be degraded.

From the literature it is evident that there is no technique for ontology matching which uses only selective concepts from the ontologies to find the degree of similarity between them. The need is to find similarities between two ontologies in an efficient way by comparing only the key concepts of the ontologies rather than comparing all important and non-important concepts. However, there are few techniques for selecting important concepts from an ontology. D'Entremont and Storey (2006) proposed to select important concepts by following the user's browsing activities. Tu *et al.* (2005) proposed to filter important concepts from an ontology based on concept hierarchy without considering non-subsumption relations between concepts. More detailed information about ontology structure, like the correlation between concepts and relations, has not been explored. In some other studies, traditional link analysis ranking algorithms are employed to find important concepts and relations (Balmin *et al.*, 2004; Ding *et al.*, 2005), but these approaches require time-consuming machine learning methods.

This paper presents an approach for finding how similar an ontology is with another ontology on the web, based on only comparing prominent concepts of the ontologies being matched.

3. Proposed approach

The main idea of the proposed approach is to reduce search space and time complexity in finding a degree of similarity between two ontologies. The complexity of matching is generally proportional to the size of the ontologies under concern. A suitable approach for minimizing the time complexity is to reduce the number of pair-wise comparisons. The proposed approach performs ontology matching by comparing the key concepts of input ontologies rather than comparing all concepts of each ontology. Along with local names of concepts, the synonym list for each key concept from WordNet (Miller, 1995), is also compared. The technique not only matches the classes of the ontologies but also the relations among the classes as well. We define the key concept of ontology as follows:

Key concept: a concept is considered to be the key concept of an ontology if it is more popular in that ontology. In other words, if it has more relations with the other concepts in the ontology, then it is a key concept of the ontology.

The technique is comprised of two types of matchers, "Class Matcher" and "Relation Matcher." There are three modules of the proposed approach Key Concept Finder, Class Matcher, Relation Matcher.

Key Concept Finder finds the key concepts of an ontology by measuring the popularity of a concept (class name or relation). Popularity measure is defined by the following equation:

$$\text{Popularity}\left(C_i,\ O_j\right) = n_{i,\,j}/\Sigma_k n_{k,\,j} \tag{1}$$

where $n_{i,j}$ is the number of occurrences of C_i in O_j; and $\Sigma_k n_{k,j}$ the number of occurrences of all concepts in ontology $O_{j.}$

The numerator in Equation (1) is computed on the basis of how many subclasses, super classes, disjoint classes and related properties a concept has, in the given ontology. The popularity measure is similar to the term frequency, a very important measure in information retrieval, used to find the important keywords from a document. The higher value of popularity for a concept shows that the concept best represents the ontology. More specifically, if $C=[C_1, \ldots, C_n]$ is the set of concepts returned by the Key Concept Finder algorithm and Z_i is a concept in the ontology, there should be a concept $C_k \in C$ such that either $Z_i \subseteq C_k$ or $C_k \subseteq Z_i$ holds. The motivation for the popularity measure is that not only the right type of concepts should be returned by the proposed method, but also the right distribution of concepts must be retrieved, to provide the best achievable representation of the ontology:

Algorithm 1: Key Concept Finder Algorithm
Input: Ontology Concepts

Output: Key Concepts
Take the input ontology in the OWL format
Declare key_concepts[ ]
For each concept C_i
  Compute Popularity (C_i)
  If Popularity (C_i) > threshold
    key_concepts [] = C_i;

After finding the key concepts from both input ontologies, they are passed to the Class Matcher and Relation Matcher. "Class Matcher" computes mapping between the key classes of both ontologies to be matched. This module uses WordNet as the external resource for finding synonyms of the key concepts, so that if two concepts being compared have different names (with similar meaning), they are evaluated to be similar.

Relation Matcher computes mapping between the key relations of both ontologies to be matched. This module uses WordNet as the external resource for finding synonyms of the key relations. Two relations are considered to be mapped if they share the same domain and range:

Algorithm 2: Class Matcher algorithm
Input: Key Classes from ontology O_i and O_j
Output: Class Mappings
  1. Initialize Class_Matches = 0;
  2. Declare class_intersection [ ][ ];
  3. For each key_class $C_{k,i}$ in O_i
      Access WordNet for synonyms(if exist any)
      $S_{k,i}$ = getSyno($C_{k,i}$);
  4. For each key_class $C_{k,j}$ in O_j
      Access WordNet for synonyms (if exist any)
      $S_{k,j}$ = getSyno($C_{k,j}$);
  5. For each $C_{k,i}$ in O_i
      For each $C_{k,j}$ in O_j
          if($C_{k,i}$ = = any of $C_{k,j}$ OR any Element of $S_{k,i}$ = = any Element $S_{k,j}$)
            class_intersection [][0] = $C_{k,i}$;
            class_intersection [][0] = $C_{k,j}$;
            Class_Matches++;

Algorithm 3: Relation Matcher algorithm
Input: Key relations from ontology O_i and O_j
Output: relation mappings
  1. Initialize rel_Matches = 0;
  2. Declare rel_intersection [ ][ ];
  3. For each key_relation $R_{k,i}$ in O_i
      $Sdomain_{k,i}$ = getSyno (domain ($R_{k,i}$));
      $Srange_{k,i}$ = getSyno (range ($R_{k,i}$));
  4. For each key_relation $R_{k,j}$ in O_j
      $Sdomain_{k,j}$ = getSyno (domain ($R_{k,j}$));
      $Srange_{k,j}$ = getSyno (range ($R_{k,j}$));
  5. For each $R_{k,i}$ in O_i
      For each $R_{k,j}$ in O_j

if(($R_{k,i}$ = = $R_{k,j}$) OR (domain ($R_{k,i}$) = = domain ($R_{k,j}$) && range ($R_{k,i}$) = = range ($R_{k,j}$)) OR (any element of $Sdomain_{k,i}$ = = any element of $Sdomain_{k,j}$ && any element of $Srange_{k,i}$ = = any element of $Srange_{k,j}$))

rel_intersection [][0] = $R_{k,i}$;
rel_intersection [][0] = $R_{k,j}$;
rel_Matches++;

4. Results and discussion

Evaluation of the proposed approach is performed with the conference data set created by the OAEI. This data set consists of seven real world ontologies describing conferences and the 21 reference alignments among them are given. The simple statistics on these ontologies are given in Table I.

The proposed approach is implemented in Java and OWLAPI and is used for navigating the input ontologies. For each pair of ontologies, their key concepts are discovered by measuring the popularity of each concept. The concepts having a popularity measure greater than the threshold of 0.002 and 0.001 are selected as key concepts. Then the key concepts of both ontologies are compared to find the degree of similarity between two ontologies. WordNet 3.0 is used for finding synonyms for the key concepts.

The degree of similarity is calculated by finding the ratio of the number of mappings found by our technique and number of key concepts in both ontologies qusing Equation (2). The same method of finding the degree of similarity is applied to the reference alignments as shown in Equation (3):

$$\text{Degree of similarity} = \frac{\text{No. of mappings found}}{\text{No. of key concepts in both ontologies}} \times 100 \quad (2)$$

$$\text{Degree of similarity (Reference Alignment)} = \frac{\text{No. of mappings found in reference alignmet}}{\text{No. of concepts in both ontologies}} \times 100 \quad (3)$$

Figure 1 shows the degree of similarity obtained from the proposed approach and the degree of similarity obtained from the reference alignments given on OAEI. It can be seen from the graph that when the degree of similarity of reference alignment increases, the degree of similarity computed by the proposed approach also increases and the same happens when the degree of similarity of reference alignment decreases.

Pearson's correlation coefficient is calculated to show that the strength of the degree of similarity obtained computed by our approach is related to the degree of similarity

Table I. Simple statistics of the data set

Ontology	No. of concepts	No. of properties
Cmt	36	59
Ekaw	74	33
Edas	104	50
Iasted	140	41
Sigkdd	49	28
Conference	59	64
confOf	38	36

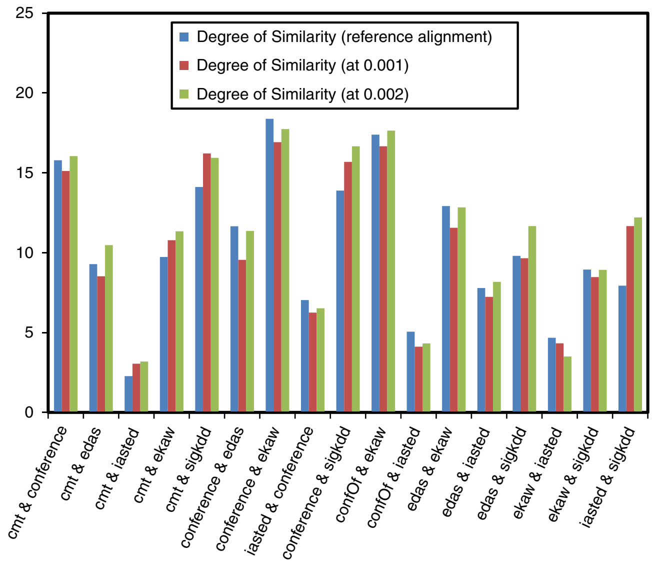

Figure 1.
Degree of similarity vs degree of similarity (reference alignment)

obtained from the reference alignment. The formula for the Pearson's correlation coefficient r is given as:

$$r = \frac{n\left(\sum xy\right) - \left(\sum x\right)\left(\sum y\right)}{\sqrt{\left[n\sum x^2 - \left(\sum x\right)^2\right]\left[n\sum y^2 - \left(\sum y\right)^2\right]}} \tag{4}$$

where n is the number of data points of x and y; x the 1st variable, this is the degree of similarity (reference alignment) here; and y the 2nd variables, this is the degree of similarity here.

Pearson's correlation coefficient can take value in the range −1 to 1. The value of r close to 1 shows that variable x and variable y have perfect positive correlation, i.e. they follow the same trend, while values close to −1 show that variable x and y have perfect negative correlation. The value of r obtained at a threshold value of 0.002 and 0.001 is 0.85 and 0.82, respectively, in our approach as shown in Figure 2, which illustrates a positive correlation between the degree of similarity and the degree of similarity (reference alignment).

This positive correlation between the degree of similarity and the degree of similarity (reference alignment) showed that if, only the key concepts of both ontologies are considered for matching purpose, the same or even better results can be achieved rather than doing an exhaustive list of comparisons between all the concepts of both ontologies.

Along with finding the degrees of similarities, we wanted to check the level of precision and relevance of the results our technique gives. Therefore, the precision, recall and F-measure were computed to check how well the technique performs. The graph in Figure 3 shows the precision, recall and F-measure observed for each pair of ontologies on the conference track OAEI.

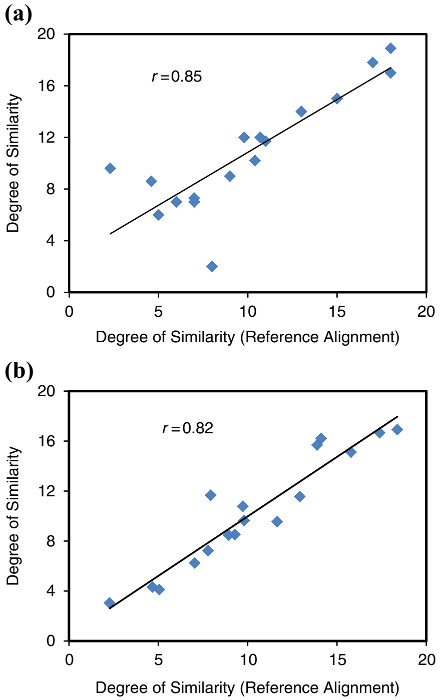

Figure 2.
Pearson's correlation coefficient

Notes: (a) 0.002; (b) 0.001

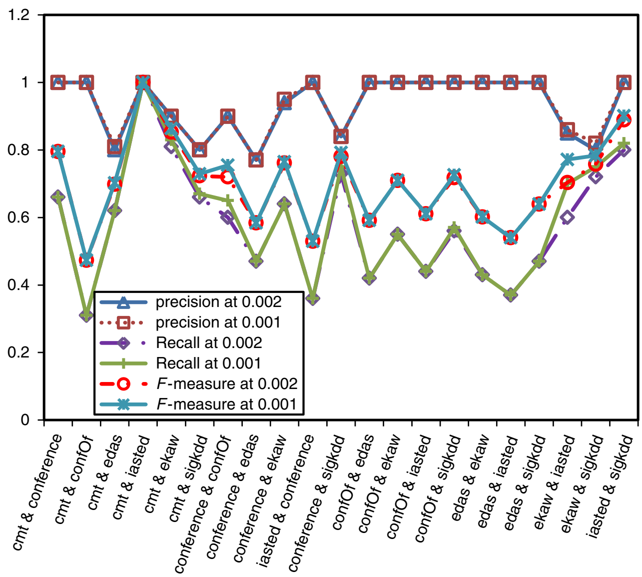

Figure 3.
Precision, recall and *F*-measure

The comparison of average precision, *F*-measure and recall of participant from OAEI 2012 and our approach is presented in Table II.

The results show that good similarity results can be obtained even by only involving prominent concepts of the input ontologies in the matching process, thus achieving good performance in less time.

For the comparison of the reduced search space and time requirements achieved by the proposed approach, a traditional ontology matching approach based on Noy and Musen (2003), Doan *et al.* (2004), Ehrig and Sure (2004), is developed and used as baseline. This traditional ontology matching approach compares all classes and properties of one ontology with all classes and properties of other ontology. The comparison is done on a regular computer system with specification (Intel® Core (TM) i3 CPU, 2.3 GHz, 2.0 GB RAM and Windows 7).

Figure 4 represents the number of entities (classes and properties) matched by both the baseline and the proposed Key Matching Approach. Figure 5 shows the time in milliseconds taken by both the baseline and the Key Matching Approach. Number of tested entities and execution time taken by both approaches is calculated for each pair of ontologies on OAEI conference track. It is evident from Figures 4 and 5 that the number of entity comparisons made and the execution time baseline approach is larger than the proposed approach.

Analysis of results for both thresholds (0.001 and 0.002) demonstrates that more concepts are selected as key concepts at 0.001. The precision value at both thresholds is the same which shows that the accuracy of the proposed approach is similar to those of fuller methods; whereas a reduced degree of similarity at the lower threshold shows that comparing increased sets of concepts only increases the time and search space requirements without improving the accuracy. The value of the Pearson correlation coefficient also reduces to 0.82 at threshold 0.001 indicating the slightly weak correlation between degree of similarity, than at threshold 0.002. From the experimental observations, it is concluded that 0.002 threshold gives the best trade-off between accuracy and efficiency.

Ontology matching techniques	Precision	Recall	*F*-measure
AgrMaker	0.53	0.62	0.58
LogMap	0.77	0.53	0.57
CODI	0.74	0.55	0.58
Optima	0.6	0.63	0.62
GOMMA	0.79	0.43	0.47
Hertuda	0.7	0.46	0.49
HotMatch	0.67	0.47	0.5
ServoMap	0.68	0.41	0.45
AROMA	0.49	0.41	0.31
YAM++	0.78	0.65	0.67
MaasMatch	0.68	0.52	0.53
LogMapLt	0.68	0.45	0.48
ServOMapLt	0.68	0.41	0.45
MEDLEY	0.83	0.45	0.42
MapSSS	0.47	0.46	0.46
AUTOMSv2	0.64	0.33	0.37
Our approach at 0.002	0.79	0.58	0.67
Our approach at 0.001	0.79	0.59	0.67

Table II. Performances of OAEI 2012 participants and key concept matching

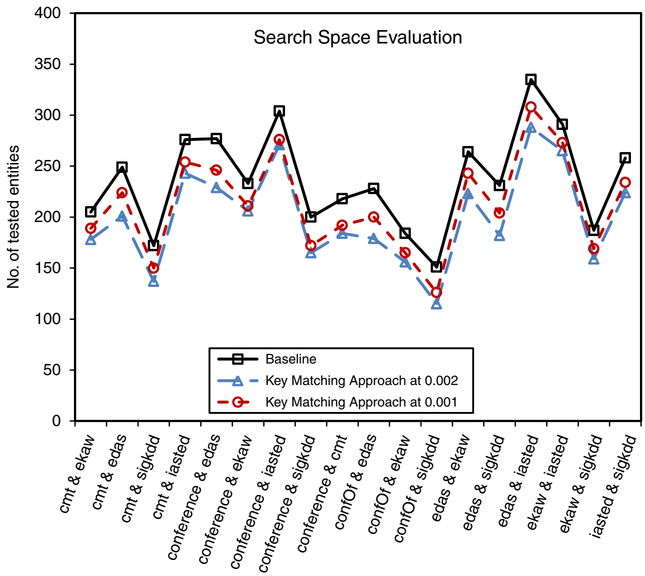

Figure 4.
Search space evaluation

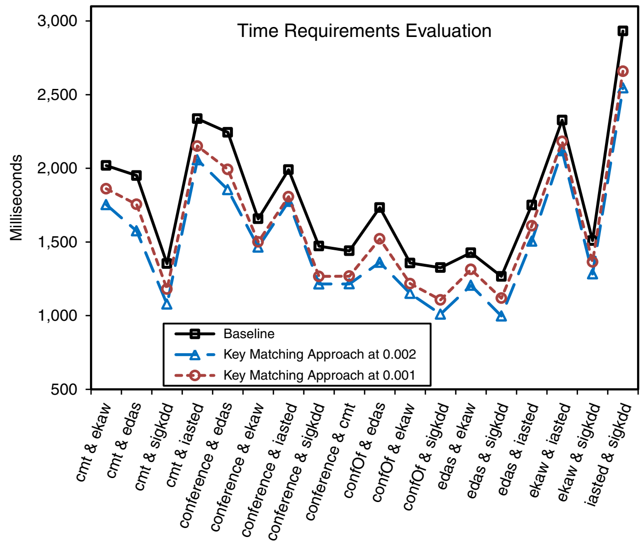

Figure 5.
Time requirements evaluation

5. Conclusion

This paper proposes a novel approach for ontology matching by selecting the key concepts from the input ontologies and then comparing those key concepts only, to find out how similar the input ontologies are with each other. It is very useful to avoid checking similarity of unmatchable concepts. The evaluation of technique shows that the same degree of similarity can be achieved by taking into account the key/representative concepts of the ontologies instead of comparing each and every concept of the input ontologies. The high values of precision, recall and *F*-measure show the enhanced accuracy by reducing mismatching concepts. The approach is a step toward solving the heterogeneity problem in less time and reduced search space. The proposed approach can be used by web query interfaces for query expansion and answering by efficiently finding underlying ontology/schema matching and to integrate large database schemas. The future aim of the research will be to embrace the merging of the two ontologies once it is concluded that the two ontologies are similar and refining the criteria of key concept selection which best captures the semantic constraints on the ontological entities.

References

Alasoud, A., Haarslev, V. and Shiri, N. (2008), "An effective ontology matching technique", *Foundations of Intelligent Systems*, Springer, Berlin Heidelberg, pp. 585-590.

Aleksovski, Z., Klein, M., Ten Kate, W. and Van Harmelen, F. (2006), "Matching unstructured vocabularies using a background ontology", *Managing Knowledge in a World of Networks*, Springer, Berlin Heidelberg, pp. 182-197.

Balmin, A., Hristidis, V. and Papakonstantinou, Y. (2004), "Objectrank: authority-based keyword search in databases", *Proceedings of the Thirtieth International Conference on Very Large Data Bases, VLDB Endowment*, Vol. 30, pp. 564-575.

Breitman, K.K., Brauner, D., Casanova, M.A., Milidiú, R., Gazola, A. and Perazolo, M. (2008), "Instance-based ontology mapping", *Fifth IEEE Workshop on Engineering of Autonomic and Autonomous Systems*, *IEEE*, pp. 67-74.

Cruz, I.F., Antonelli, F.P. and Stroe, C. (2009), "AgreementMaker: efficient matching for large real-world schemas and ontologies", *Proceedings of the VLDB Endowment*, Vol. 2 No. 2, pp. 1586-1589.

D'Entremont, T. and Storey, M.-A. (2006), "Using a degree-of-interest model for adaptive visualizations in Protégé", *9th International Protégé, Conference Citeseer*, p. 34.

Ding, L., Pan, R., Finin, T., Joshi, A., Peng, Y. and Kolari, P. (2005), "Finding and ranking knowledge on the semantic web", *The Semantic Web – ISWC 2005*, Springer, Berlin Heidelberg, pp. 156-170.

Do, H.-H. and Rahm, E. (2007), "Matching large schemas: approaches and evaluation", *Information Systems*, Vol. 32 No. 6, pp. 857-885.

Doan, A., Madhavan, J., Domingos, P. and Halevy, A. (2004), "Ontology matching: a machine learning approach", in Staab, S. and Studer, R. (Eds), *Handbook on Ontologies*, Springer, Berlin Heidelberg, pp. 385-403.

Ehrig, M. and Sure, Y. (2004), "Ontology mapping – an integrated approach", *The Semantic Web: Research and Applications*, Springer, Berlin Heidelberg, pp. 76-91.

Euzenat, J., Isaac, A., Meilicke, C., Shvaiko, P., Stuckenschmidt, H., Svab, O., Svatek, V., Hage, W.R.V. and Yatskevich, M. (2007), "First results of the ontology alignment evaluation initiative 2007", *2nd International Workshop on Ontology Matching (OM) at the International Semantic Web Conference (ISWC) and Asian Semantic Web Conference (ASWC)*, *Busan*, pp. 96-132.

Euzenat, J., Ferrara, A., Hollink, L., Isaac, A., Joslyn, C., Malaisé, V., Meilicke, C., Nikolov, A., Pane, J. and Sabou, M. (2009), "Results of the ontology alignment evaluation initiative 2009", *Proceedings of 4th ISWC Workshop on Ontology Matching (OM)*, pp. 73-126.

Gligorov, R., ten Kate, W., Aleksovski, Z. and Van Harmelen, F. (2007), "Using Google distance to weight approximate ontology matches", *Proceedings of the 16th International Conference on World Wide Web, ACM*, pp. 767-776.

Hertling, S. and Paulheim, H. (2012), "WikiMatch – using Wikipedia for ontology matching", *Proceedings of the 7th International Workshop on Ontology Matching, Citeseer*, pp. 37-48.

Hu, W. and Qu, Y. (2008), "Falcon-AO: a practical ontology matching system", *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol. 6 No. 3, pp. 237-239.

Hu, W., Qu, Y. and Cheng, G. (2008), "Matching large ontologies: a divide-and-conquer approach", *Data & Knowledge Engineering*, Vol. 67 No. 1, pp. 140-160.

Jain, P., Yeh, P.Z., Verma, K., Vasquez, R.G., Damova, M., Hitzler, P. and Sheth, A.P. (2011), *Contextual Ontology Alignment of Lod with an Upper Ontology: A Case Study with Proton*, Springer, Berlin Heidelberg.

Jiménez-Ruiz, E. and Grau, B.C. (2011a), "LogMap: logic-based and scalable ontology matching", *Lecture Notes in Computer Science*, Vol. 7031, pp. 273-288.

Jiménez-Ruiz, E. and Grau, B.C. (2011b), *Logmap: Logic-Based and Scalable Ontology Matching*, Springer, Berlin Heidelberg.

Lee, M.L., Yang, L.H., Hsu, W. and Yang, X. (2002), "XClust: clustering XML schemas for effective integration", *Proceedings of the Eleventh International Conference on Information and Knowledge Management, ACM*, pp. 292-299.

Li, J. (2004), "LOM: a lexicon-based ontology mapping tool", DTIC Document.

Lin, F. and Krizhanovsky, A. (2011), "Multilingual ontology matching based on Wiktionary data accessible via SPARQL endpoint", arXiv preprint arXiv:1109.0732.

Lin, F., Butters, J., Sandkuhl, K. and Ciravegna, F. (2010), "Context-based ontology matching: concept and application cases", *2010 IEEE 10th International Conference on Computer and Information Technology (CIT), IEEE*, pp. 1292-1298.

Mao, M. (2008), "Ontology mapping: towards semantic interoperability in distributed and heterogeneous environments", ProQuest.

Marcos Martínez, R., Manuel Vázquez, J.N., Javier Pereira, L. and Norberto, E. (2009), "Ontology alignment techniques", in Rabuñal Juan Ramón, D., Julian, D. and Alejandro, P. (Eds), *Encyclopedia of Artificial Intelligence*, IGI Global, Hershey, PA, pp. 1290-1295.

Mascardi, V., Locoro, A. and Rosso, P. (2010), "Automatic ontology matching via upper ontologies: a systematic evaluation", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22 No. 5, pp. 609-623.

Miller, G.A. (1995), "WordNet: a lexical database for english", *Communications of the ACM*, Vol. 38 No. 11, pp. 39-41.

Mork, P. and Bernstein, P. (2004), "Adapting a generic match algorithm to align ontologies of human anatomy", *Proceedings 20th International Conference on Data Engineering, IEEE*, pp. 787-790.

Ngo, D. and Bellahsene, Z. (2012), "YAM++: a multi-strategy based approach for ontology matching task", *Knowledge Engineering and Knowledge Management*, Springer, Berlin Heidelberg, pp. 421-425.

Noy, N.F. and Musen, M.A. (2003), "The PROMPT suite: interactive tools for ontology merging and mapping", *International Journal of Human-Computer Studies*, Vol. 59 No. 6, pp. 983-1024.

Paulheim, H. (2008), "On applying matching tools to large scale ontologies", *OM-2008 Held in Conjunction with ISWC*, pp. 214-218.

Sabou, M., d'Aquin, M. and Motta, E. (2008), "Exploring the semantic web as background knowledge for ontology matching", *Journal on Data Semantics XI*, Springer, Berlin Heidelberg, pp. 156-190.

Saleem, K., Bellahsene, Z. and Hunt, E. (2008), "Porsche: performance oriented schema mediation", *Information Systems*, Vol. 33 No. 7, pp. 637-657.

Seddiqui, M.H. and Aono, M. (2009), "An efficient and scalable algorithm for segmented alignment of ontologies of arbitrary size", *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol. 7 No. 4, pp. 344-356.

Shvaiko, P. and Euzenat, J. (2008), *Ten Challenges for Ontology Matching*, Springer, Berlin Heidelberg.

Shvaiko, P. and Euzenat, J. (2013), "Ontology matching: state of the art and future challenges", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25 No. 1, pp. 158-176.

Tu, K., Xiong, M., Zhang, L., Zhu, H., Zhang, J. and Yu, Y. (2005), "Towards imaging large-scale ontologies for quick understanding and analysis", *The Semantic Web – ISWC 2005*, Springer, Berlin Heidelberg, pp. 702-715.

Corresponding author
Tayybah Kiren can be contacted at: tayybah@uet.edu.pk