Journal of Theoretical and Applied Information Technology
20th April 2015. Vol.74 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ONTOLOGY MATCHING: IN SEARCH OF CHALLENGES AHEAD

[1,3]FAROOQUI N.K, [2]Dr. MOHAMMED FAUZAN NOORDIN AND [2,4]ABDULHAFEEZ MUHAMMAD

[1]Department of Computer Science, Kulliyah of Information Communication Technology, International Islamic University (IIUM), Kuala Lumpur, Malaysia.
[2]Department of Information Systems, Kulliyah of Information Communication Technology, International Islamic University (IIUM), Kuala Lumpur, Malaysia.
[3]Department of Information Systems, College of Computer Engineering and Sciences, Salman Bin Abdulaziz University, Alkharj, Saudi Arabia.
[4]College of Computer Science, King Khalid University, Kingdome of Saudi Arabia.

Email: [1]farooquinajam@gmail.com/farooqui.najmuddin@live.iium.edu.my, [2]fauzan@iium.edu.my, [3]abdul.hafeez@live.iium.edu.my/ abdulhafeez97@gmail.com

ABSTRACT

This paper presents key features and challenges for the development of the next generation ontology matcher. Matching elements of two data instances plays an important role in e-business, multilingual data instances, biomedical and open data cloud. This paper elaborates technologies, tools, algorithms and methods used by recent ontology matchers. Ontology matching has become a Meta Research field where research topics such as developing advanced reasoner, algorithms for optimum matching, working on Meta matchers and improving results are major open research areas. In order to find the future direction towards the development of optimum matchers we illustrated a list of future challenges, key features, and their importance. This paper does not propose solution or framework for an optimum matcher. Reader will get help in deciding next ontology matching techniques in his domain of ontology matching research.

Keywords*:* *Context, Knowledge Management, Ontology matching, Semantic web, Semantic technologies, Semantic literature review.*

1. INTRODUCTION

Semantic and knowledge engineering researchers are focusing on matching ontologies. In order to convert the digital knowledge bases into semantic knowledge, researchers and IT practitioners have developed large ontologies and even made it available on Linked Open data.
Ontology matching is a research area that focuses on knowledge discovery using matching ontologies. Ontology matching technologies focus on various techniques, tools, reasoner and interfaces.
Ontology matching systems such as LogMap (Grau, 2011) and Blooms and Blooms+ (Jain, Hitzler, Sheth, Verma, &Yeh, 2010), AgreementMakerLight (Faria et al., 2013) have developed some recent matching systems. It is important to study key feature of these popular ontology matching systems and the challenges in order to develop a comprehensive next generation ontology matching system.
Several decades of information technology development has resulted in a huge amount of information generating consequence is the mismatch and inefficient /incomplete data. Now, information technology community is working towards integration of these information resources. Semantic technology is one of them which helps in retrieving meaningful content from the information stored.
An ontology typically provides a vocabulary that determine a domain of interest and a specification of the meaning of the term used in the vocabulary.(Shvaiko, 2013)
Ontology matching is the process of matching or aligning two input ontologies (one source ontology and one target ontology) consists in finding semantic relationships between the classes of the source ontology and the classes of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the target ontology. (Faria et al., 2013)
It is the solution to information heterogeneity problem. It finds correspondence between semantically related components of ontologies. These corresponding enables us for merging query answering or data translation from one system to another system. Matched ontology and data expressed in the matched ontologies can inter operate.
Many solutions are already such asAgreementMakerlight, (Faria et al., 2013), WikiMatch(Sven Hertling, 2012), BLOOMS+(Jain et al., 2011) and LogMap(Grau, 2011)also a book published on Ontology Matching (J. Euzenat & Shvaiko, 2007).
Purpose of this research is to find the possible research areas in ontology matching, particularly focusing on methodologies and techniques for developing next generation matchers.
Such a study is required to sort specifically out current challenges in detail with specific and sub research areas. In order to elaborate clear and concise idea about open research issues in ontology matching, we have categorized it into three different groups (See Figure 4).
Objectives of this work are to study recently developed ontology matching systems. We will analyze methods and techniques used by researchers in each system. In addition, we list key findings and features in these systems (see Table 1). In results and discussion, we list the challenges ahead for developing advanced ontology matching tool

1.2 Related Works

A comprehensive and detailed survey of this field was presented by (Otero-Cerdeira, Rodríguez-Martínez, & Gómez-Rodríguez, 2014). They presented a complete picture of the current status of research on this subject. Efficient and detailed categorizations of almost all publications are done. A survey study of semantic integration (Noy, 2009) provide a brief introduction. This paper presents different themes in semantic integration field and points to different project in these areas. Reasonerare compared by (Kathrin Dentler a, b,∗, Ronald Cornet a, 2011) on different parameters such as characteristics, usability and performance indicators. (P. S. and J´. Euzenat, 2005)Presented schema based ontology matching survey. They compared various features and combination of matching at schema level with matching properties such as input information their properties such as syntactic, semantic, element level or structure level. Ontologies and Ontology matching are the requirements of efficient knowledge management. (Noordin, Othman, & Zakaria, 2011) discuss the importance of the knowledge worker in implementing knowledge management systems. (Rahm & Bernstein, 2001)Presented survey of tools using automatic schema matching techniques. They categorize them according to the schema level, instance level, element level, and language based or constraint-based. (Dong, Hussain, & Chang, 2009) Presented a novel ontology-based web page classification method for the knowledge grid environment. They used web page metadata to classify the web pages by ontology concepts.

1.3 Limitation ofThis Study

This study covers study of ontology matching system considering the factors as discussed in section 2 most important of which are two factors on which this study is based on firstly general knowledge source as source ontologies and secondly robust and applied system that may have developed over years and become mature over consistent development every year.

2. RECENT ONTOLOGY MATCHING SYSTEMS

We have selected ontology matching system in such a way that, they cover almost all areas of ontology matching system. The selected systems are recent, robust, extensible and known to ontology matching community. Here we discuss in detail, techniques used by those ontology matching tools.

Research Questions

We put following research questions to find the optimum matchers and its technique.

Q.1.What are the important parameters affecting the performance of matching system?
Q.2. Which are the open research areas in this field?
Q.3. What is major ontology matching system?
Q.4. What are major sources of ontologies used in matching system?
Q.5. How to present matching system results such as filtering duplicates?

Following are the factors that we have considered the selection of matching system.
1. Matching system that have consistently

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

developed over years or they become robust.
2. Matching system that considers matching proper source of knowledge such as Wikipedia.
3. Matching systems that are context oriented.
4. Matching systems that are considering popular tools such as protégé.

2.1 AgreementMakerLight Ontology Matching System (Faria et al., 2013)

AgreementMakerlight is one of the leading ontology matching systems thanks to its combination of a flexible and extensible framework for a comprehensive user interface. It is focused on computational efficiency and designed to handle very large ontologies. Figure 1 shows AgreementMakerLight framework. The source ontology block is ontology loader, which uses Jena2 ontology API.

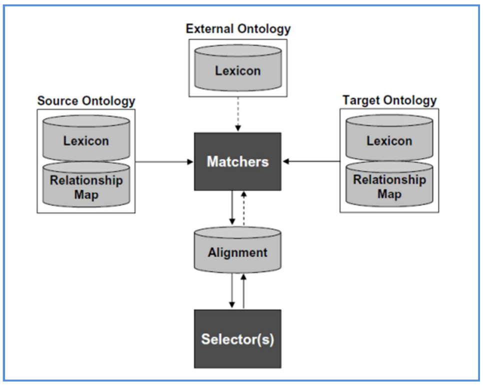

Figure 1:AgreementMakerLight.

In order to reduce, the processing time especially for large ontologies, AgreementMakerLight uses internal data structures. ***Lexicon*** is a data structure that links each class in the ontology with its names. ***RelationshipMap*** is a data structure that links, each class to the classes related to it through *is an* or *part of* relationship or disjoint clauses. ***Alignment*** is a data structure used by the ontology matching module to store mappings between the input ontologies. During the matching procedure, the primary data structure used by AgreementMaker is a matrix that stores the similarities between all concepts of the source ontology against all concepts of the target ontology.

Ontology matching module contains three components, *Matchers*(i.e. match algorithms), *Selectors* (i.e. selection algorithms) and previously described as *Alignment data Structure* as shown in figure. Matches are divided into two parts, primary and secondary matchers. Selectors are algorithms used to trim an alignment by excluding mappings below a given similarity threshold and excluding competing mappings to obtain the desired cardinality. There are three implemented matchers*Lexical Matchers, Mediating Matchers, and word Matchers.*

2.2 An Efficient and Scalable Algorithm For Segmented Alignment Of Ontologies Of Arbitrary Size (Anchor Flooding) ((Seddiqui & Aono, 2012)

This algorithm starts with an anchor, a pair of 'Look a Like' concepts from each ontology, gradually exploring concepts by collecting neighboring concepts. They focus on the segment to segment comparison. Main processing blocks of the algorithm contains three mutual exploring steps i.e. exploring alignment concepts (EA), exploring unaligned concepts (UC) and exploring parents (EP). They applied anchor flood algorithms to a variety of datasets. They discussed results with factors such as precision and recall, time efficiency memory efficiency, scalability, segmented alignment and complexity analysis.

2.3 WikiMatch(Sven Hertling, 2012)

WikiMatch is an Ontology alignment system that like BLOOMS+ uses Wikipedia as an external knowledge source. Compared to BLOOMS+, however, WikiMatch's method is simpler and takes less runtime. Wikipedia contains approximately31,000,000 articles that cover most every possible domain, though at varying levels of depth. These articles are written by volunteers around the world in 285 languages. Because Wikipedia articles exist in different natural languages, and the same article in different languages are linked, WikiMatch can also be used to align ontologies in different natural languages.

For each ontology concept, Wikimatch extracts the following components: fragment, labels and comments in a string format. The Wikipedia search engine is used to search for these strings with stop-words removed. For each concept, WikiMatch generates a set of document ids for URI fragment Sf, a set of documents ids for label Sl, and a set of document ids for a comment Sc.From those sets, the similarity of source concepts and a target concept is computed by

Journal of Theoretical and Applied Information Technology
20th April 2015. Vol.74 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

maxS in _{Sf,Sc,Sl}=#(S s ∩S t)#(S sUS t) (4)

Where *S*(s) and *S*(t) is the set of document ids for each of the three respective components for ten the source and target concept, respectively. The similarity of the source concept and target concept is derived from the similarity of their sets of documents for each of the three components. If the maximum similarity of those three components exceeds a certain threshold, it returns an equivalence mapping between the source and target concepts. WikiMatch uses two Wikipedia searches. The simple search approach uses the complete string and not the individual tokens in the component. WikiMatch also implements another approach called the individual token search approach where each token in the component is used to search Wikipedia.

Wikimatch differs from BLOOMS+ in that it does not build a category tree but takes a much simpler approach and uses the set intersection and union on the document ids returned by the Wikipedia search engine for each concept. These sets are determined based on which component, i.e., fragment, label and comment are used to create the search strings passed to the Wikipedia search engine.

2.4 BLOOMS And BLOOMS+

Bootstrapping-based Linked Open Data Ontology Matching (*BLOOMS*) system (Jain et al., 2010) and BLOOMS+ (Jain et al., 2011) are Ontology alignment systems for automatically finding correspondences between Linked open data ontologies. Both uses Wikipedia as a knowledge source to construct a set of category hierarchy trees called a *forest* for each concept in the source and each concept in the target ontology. Both BLOOMS and BLOOMS+ use the same procedure to build the category trees. Dealing with multiple pages, they considered a root node and built a tree around it.

2.5 LogMap Overview

LogMap (Grau, 2011) implements highly optimal data structures for both lexically and structurally indexing the input ontologies. These structures are used to compute an initial set of anchor mappings and to assign a confidence value to each of them. The core of LogMap follows the general architecture for Ontology alignment systems in that it uses an iterative process that starts from the initial anchors and then alternates the mapping repair and mapping discovery steps. In order to detect and repair unsatisfiable classes (Determining by propositional Horen representation) `on the fly' during the matching process, LogMap implements a sound and highly scalable ontology reasoner as well as a `greedy' diagnosis algorithm. New mappings are discovered by iteratively `exploring' the input ontologies starting from the initial anchor mappings and using the extended hierarchies of the two input ontologies.

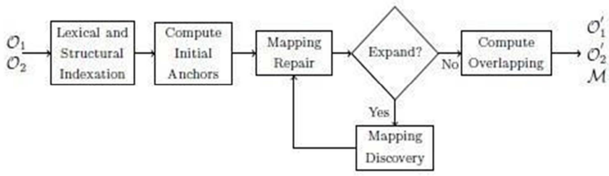

Figure 2: LogMap in nutshells (Grau, 2011)

The main steps of LogMap process are as following:

(1) Lexical and structural indexation**:** LogMap indexes the labels of the classes in ontology as well as their lexical variations and allows for the possibility of enriching the indexes by using an external lexicon (e.g., WordNet or UMLS-lexicon). For structural indexing, LogMap uses an interval labeling schema to represent the extended class hierarchy of the input ontology. Each extended hierarchy can be computed using either simple structural heuristics or a DL reasoner.

(2) Computation of initial `anchor mappings'**:** LogMap computes an initial set of equivalence anchor mappings by intersecting the lexical indexes of the input ontology. These mappings can be considered "exact" and later serve as starting point for the further discovery of additional mappings.

(3) Mapping repair and discovery**:** In the repair step, LogMap uses a reasoning algorithm to detect classes that are unsatisfiable for both input ontologies and the current mappings computed so far. Then, each of these undesirable logical consequences is automatically repaired using a `greedy' diagnosis algorithm that recursively repair relevant classes by using the reasoning algorithm to detect unsatisfiable classes. For discovering new mapping, LogMap maintains two contexts (One per the ontology, consist of a

Journal of Theoretical and Applied Information Technology
20th April 2015. Vol.74 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

set of classes determined by anchor and a subset of specific active classes for each iteration) for each anchor. Contexts for the same anchor are expanded in parallel using the class hierarchies of the input ontologies by new mappings are then computed by matching the classes in the relevant contexts by a flexible tool that computes a similarity score for any pair of input strings.

(4) Ontology is overlapping estimation**:** In addition to the final set of mappings, LogMap computes a fragment of the input ontology, which intuitively represents the `overlapping' between both ontologies. When manually looking for additional mappings that LogMap might have missed, curators can restrict themselves to these fragments since `correct' mappings between classes not mentioned in these fragments are likely to be rare.
The system has high level of maturity as in the last 3 years there have been at least 6 publications (Ernesto Jim´ enez-Ruiz, Ant´ on Morant, 2011)(literature review paper ref)describing its performance and results in the OAEI contests (Grau, 2011)(Jim,´, & enez-Ruiz, Bernardo Cuenca Grau, 2012)(Jim´enez-Ruiz, 2013)

2.6 Mastro: A Reasoner for Effective Ontology-Based Data Access ((Lukasiewicz, 2011)
It is a system for ontology based database access 2011. Mastro is a java tool for ontology-based data access. They used descriptive logic. JDBC connects the tool with database with back end.

Figure 3: Mastro Architecture

They developed an API and extension for Protégé 4 ontology editor. As shown in figureMastro, has different modules. QuOnto is a reasoner; it provide services such as concepts satisfy ability and concept subsumption. Mapping processor performs query unfolding.

2.7 S-Match (Fausto Giunchiglia, 2012)
Open source framework for matching lightweight ontologies. S- Match provides multiple interface and possibility of adding customized background knowledge. First they used neutral language label of the node using linguistic oracle (WorldNet) in order to understand the intended meaning. Secondly they compute meaning of the node depending on its position on the tree. Third, they find the relation between the concepts of the label of two input trees. They use both semantic and semantic matcher. Contribution of S- the Matcher is its multiple interfaces that include Java API, Command line interface, and GUI interface.

2.8 Similarity Flooding (Melnik & Garcia-molina, 2002)
This algorithm takes two graphs as input and produces output between corresponding nodes of the graph. They put filters, checked by humans. The evaluation is calculated as no of corrections required after mapping. Similarity flooding considers that, if any nodes found to be similar, adjacent nodes may be similar, or their chances of being same may be more. Using fix point computations, a graph in similarity propagation is matched. Filters work in three steps and use a bipartite graph to filter duplicate matching. Evaluation of the method is done as the number of adjustments required after mapping.

3. RESULTS

Following we key findings and challenges for new researchers.

Key Findings
Above studies on Ontology matching consider various factors. These tools use different technologies tools and apply different methods. Some matching tools use multiple matchers as one matcher cannot give desired results.

Table 1 summarizes the key finding in ontology matching systems discussed above. It elaborates on the detailed methodology and key contribution of ontology matching systems. For an example system like Mastro, provide us a new technique of knowledge management. It matches ontology with the databases. This technique has an innovative approach as the primary source of knowledge retrieval is ontology because; another side of matching is a database. This system can retrieve knowledge with semantic technique from the database. More detailed description of above systems is discussed in Section III.

Journal of Theoretical and Applied Information Technology
20th April 2015. Vol.74 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. DISCUSSION

From above studies, it is clear that, ontology matching techniques are improving and still there are open research areas. Here we discuss challenges for the researchers in coming years.

4.1 Challenges:

Although recent research on ontology matching has gain considerable result, still there are various challenges. Considering single or multiple matcher and achieving optimal results out of their combination is a challenge. In some cases, there are duplicates in matching results, which presents ambiguous results.

Above discussion, concludes that, if there has to be a perfect matching tool one need to address the current challenges. Reader will be able to sort out his way to find his or her way to developing next suitable matching tool.

Challenge 1: Source ontologies

- Selecting or making ontologies be matched is a challenge because ontologies are application dependent.
- Automatic ontology creation tools are rare and they still in the development stage.
- Ontology reuse can solve this problem to some extent. Linked open data and Dbpedia are good for such ontologies.
- Currently, matching Wikipedia/ DBpedia is on the focus as a large number of articles one great knowledge assets.
- Important area whether one source ontology can be matched with the database. Advantage of this approach is to retrieve database knowledge with the help of ontology that is better in terms of knowledge management.
- Apply different methods. Some matching tools use multiple matchers as one matcher cannot give desired results. DBpedia or Linked Open Data is a choice of matchingsystems, as they provide a good opportunity to reuse ontologies.
- It decides your working area/domain of ontologies.

Challenge 2: Algorithm development /Customization

- Achieving optimization between factors such as using multiple matchers, using lexical/structural or both techniques. Applying optimum number of synonyms for both ontologies.
- Ontology matching systems such as anchor flooding uses algorithms. They apply greedy algorithms and dynamic project.Matching required customization and modified of source ontologies.

Challenge 3: Application of reasoner.

- Applying rule-based logic in the reasoner is a good challenge/opportunity, because most of the logical statement in ontology matching requires descriptive logic. Using descriptive logic makes matcher complex.
- Description logic in the reasoner is common in ontology matching.
- Effective definition of rules for matching ontologies needs to define in the reasoner, which is a tough task as specific definitions that suits all condition are difficult.

Challenge 4: Filtering and removing duplicate results

- Fine tuning results so that it becomes meaningful and applicable.
- Matching ontologies results in duplicate matching.
- Using matching techniques can remove duplicates.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Matching System	Tools	Techniques Used	Source of Ontologies	Multilingual	Key Features
AgreementMaker 2009	-----	1.Lexical relationship alignment 2.Matchers selectors alignment data structure 3.Use Multiple Algorithms	Any type of ontology	No	1.Multiple Matchers 2. User Linear weighted combination to calculate matching. 3. User can use multiple matching combinations as per requirements; this brings flexibility.
BLOOMS and BLOOMS+	Java	Upper-level ontology.	Wikipedia /DBpedia ontologies		1. Linked open data ontology matching. 2. Use upper ontology to add context.
LogMap 2013	Developed in Java	1.Both lexical and structural Matching 2. Three-step solutions including repairing step	Any ontology	No	1. Iterative in nature 2. Multiple matchers 3. Use both Algorithms structural and lexical matching algorithms
Mastro 2011	Java	1.Use Descriptive Logic 2.Process of Query unfolding is interesting	Any type of ontology mapped with RDBMS contents	No	1. API for protégé 4. 2. Connecting with database in ontological form bring lot of knowledge in ontological form.
S-Match	Java, Protégé 4 plug-in	1.JDBC Protégé 2. Possibility of adding background knowledge with every match.	Any type of ontologies with DB /RDBMS	N/A	1.Mapping with the Database can bring huge knowledge to ontological style of Knowledge management 2. Multiple interface including command line, API and GUI
Similarity Flooding 2002	----	1.Mapping corresponding nodes of graph 2. Evaluate on number of corrections required after the match. 3.iterative in nature	Any type of ontology	No	1. Assumes that if match occurs, nearest nodes will have maximum possibility of being a match.
Wikimatch 2012	Not Available online	1.Wikipedia search engine 2.Use set theory	Wikipedia /DBpedia ontologies	Yes	Use of Wikipedia / DBpedia

Table 1: Key Features Of Ontology Matching Systems

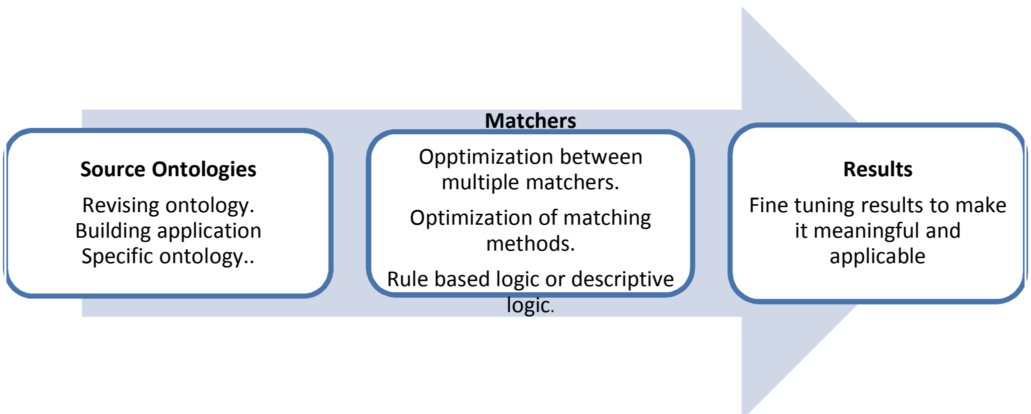

Figure 4: Challenges In The Next Generation Ontology Matching Development

Journal of Theoretical and Applied Information Technology
20th April 2015. Vol.74 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5. CONCLUSION

Depending on the study done above we conclude that following are major areas of research in ontology matching researches shown in figure 4.

1. Source ontologies
2. Matcher optimization
3. Fine tuning results

First part of our discovery where ontologies to be matched are selected. Current open research areas include sourcing ontologies from DBpedia, linked opened data or a hybrid approach where ontology is considered as source of basic concept and another side of matching is a database or webpages.

Secondly there is a plenty of space for development in developing matcher. The level of customization and application of this method to a different domain can bring new results.

Thirdly, once the matching results are achieved they include duplicates and unwanted results. Fine tuning these results is another area, where we can improve further. Fine tuning the results are more important in cases where more open-ended source ontologies are matched such as DBpedia.

Recent matching systems used iterative methods to rectify the results. These more iteration means more processing as the ontology size raises iterative method will become ineffective. There is a need to have robust method to retrieve the results.

REFERENCE:

[1] Grau, E. J.-R. and B. C. (2011). LogMap : Logic-Based and Scalable Ontology Matching. In *The Semantic Web–ISWC* (pp. 273–288).

[2] Jain, P., Hitzler, P., Sheth, A. P., Verma, K., & Yeh, P. Z. (2010). Ontology Alignment for Linked Open Data. In *The International Semantic Web Conference (ISWC)* (pp. 402–417).

[3] Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F., & Couto, F. M. (2013). The AgreementMakerLight Ontology Matching System. *Springer-*, 527–541. Retrieved from http://ceur-ws.org/Vol-1111/om2013_proceedings.pdf#page=110

[4] Shvaiko, P. (2013). Ten challenges for ontology matching ome Euzenat To cite this version : Ten Challenges for Ontology Matching.

[5] Sven Hertling, H. P. (2012). WikiMatch – Using Wikipedia for Ontology Matching. In *Ontology Matching OM-2012* (pp. 37–48)

[6] Jain, P., Yeh, P. Z., Verma, K., Vasquez, R. G., Hitzler, P., & Sheth, A. P. (2011). Contextual Ontology Alignment of LOD with an Upper Ontology : A Case Study with Proton. In *The Semantic Web: Research and Applications. Springer Berlin Heidelberg.* (pp. 80–92).

[7] Euzenat, J., & Shvaiko, P. (2007). *Ontology Matching* (p. 30). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-38721-0

[8] Otero-Cerdeira, L., Rodríguez-Martínez, F. J., & Gómez-Rodríguez, A. (2014). Ontology Matching: A Literature Review. *Expert Systems with Applications*. doi:10.1016/j.eswa.2014.08.032

[9] Noy, N. F. (2009). Semantic Integration : A Survey Of Ontology-Based Approaches. *ACM Sigmod Record*, *33*(4), 65–70

[10] Kathrin Dentler a, b,□, Ronald Cornet a, A. ten T. b and N. de K. (2011). Comparison of Reasoners for large Ontologies in the OWL 2 EL Profile. *SemanticWeb*, *2*(2), 71–87. doi:10.3233/SW-2011-0034

[11]Euzenat, P. S. and J´. (2005). A Survey of Schema-based Matching Approaches. *Journal on Data Semantics IV. Springer Berlin Heidelberg, IV*(October 2004), 146–171.

[12] Noordin, M. F., Othman, R., & Zakaria, N. A. (2011). Peopleware & heart ware - The philosophy of knowledge management. In *2011 International Conference on Research and Innovation in Information Systems* (pp. 1–6). Ieee. doi:10.1109/ICRIIS.2011.6125675

[13]Rahm, E., & Bernstein, P. a. (2001). A survey of approaches to automatic schema matching. *The VLDB Journal*, *10*(4), 334–350. doi:10.1007/s007780100057

[14] Dong, H., Hussain, F. K., & Chang, E. (2009). An ontology-based webpage classification approach for the knowledge grid environment. *SKG 2009 - 5th International Conference on Semantics, Knowledge, and Grid*, 120–127. doi:10.1109/SKG.2009.69

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[15] Jim, E.,´, & enez-Ruiz, Bernardo Cuenca Grau, and I. H. (2012). On the Feasibility of Using OWL2 DL Reasoners for Ontology Matching Problems. In *Proceedings of the OWL Reasoner Evaluation Workshop* (Vol. 858, pp. 99–105).

[16] Seddiqui, H., & Aono, M. (2012). An Efficient and Scalable Algorithm for Segmented Alignment of Ontologies of Arbitrary Size. *Web Semantics: Science, Services And Agents On The World Wide Web, 7(4).*, *7*(4), 1–34.

[17] Jim´enez-Ruiz, E. (2013). Results of the Ontology Alignment Evaluation Initiative 2013. In *Proc. 8th ISWC workshop on ontology matching (OM)* (pp. 61–100).

[18] Ernesto Jim´ enez-Ruiz, Ant´ on Morant, and B. C. G. (2011). Ontology Matching. In *Proceedings of the ISWC workshop* (p. 173).

[19] Lukasiewicz, T. (2011). The MASTRO system for ontology-based data access, *2*, 43–53. doi:10.3233/SW-2011-0029

[20] Fausto Giunchiglia, A. A. and J. P. (2012). S-MATCH :An open source framework for matching lightweight ontologies. *Semantic Web*, *3*(June), 307–317. doi:10.3233/SW-2011-0036

[21] Melnik, S., & Garcia-molina, H. (2002). Similarity Flooding : A Versatile Graph Matching Algorithm and its Application toSchema Matching. In *18th International Conference . IEEE.* (pp. 117–12).

.
.