Buletin Teknik Elektro dan Informatika
(Bulletin of Electrical Engineering and Informatics)
Vol. 2, No. 3, September 2013, pp. 182~189
ISSN: 2089-3191

Semantic Constraints Satisfaction Based Improved Quality of Ontology Alignment

Fatemeh Fakhar
Department of computer science, Faculty of engenieering, Payame Noor University, Ahvaz, Iran
e-mail: fakhar.mshd@gmail.com

Abstract

Development of informative and telecommunication technologies have caused to create much dissimilar information. As well with growing different information resources in ontology designs, the importance of management these dissimilar resources has increased. In spite of most matchers use diverse measures for discovery the mappings, some semantic inconsistencies in final alignment are unavoidable. So it is essential to enhance a post-processing phase to training error patterns in the final alignment. The impartial of this research was refining the ontology semantic constraints over defining semantic constraints by a different measure for suitable weighting to the constraints. The outcomes indicated that the standard evaluation measures better in the suggestive method and comparing with other top ranked matchers the used method can create enhanced outcomes.

Keywords: *ontology mapping, ontology alignment, semantic inconsistency, semantic verification, constraint satisfaction problem*

1. Introduction

On the Semantic Web, data are inevitably derived from much different ontology and without knowing the semantic mappings among them; information processing across ontologies is not possible. Manually discovery of such mappings are tedious, error-prone and clearly not possible at the Web scale. Therefore, different solutions have been proposed for automating the process of monitoring and integration of distributed data sources. Among the solutions proposed in field of information resources integration, ontology matching in semantic technology has attracted much attention [1].

In recent decades, a large number of ontologies in various fields of computer science, such as knowledge management, information retrieval, multimedia, software engineering, and Web services are created [1]. Unfortunately, ontologies themselves are heterogeneous and distributed. Defined by different organizations or by different people in the same organization, ontologies can have greatly different characteristics. Particularly, entities (including concepts, relations or instances) with no different meaning may have different labels in different ontologies; identical labeles may represent different meanings. Therefore, in order to reach semantic interoperability across ontologies, it is required to find out the alignment across ontologies [2]. There are some inconsistencies in the output alignment mapping tools that avoid achieving an accurate and optimal result [3-4]. Therefore identifying the constraints of the ontology and applied the matching process can improve the results. There are several mapping tools and consistent satisfying discussion on some of them such as GLUE [5], RiMOM [2], ASMOV [3], Lily [6] and PRIOR+ [7] are investigated.

Five expressions used in the present study are defined as follow [4]:

Definition 1: Ontology

Ontology is a formal specification of a shared conceptualization.

Definition 2: Matching Process

Ontology matching aims at finding correspondences between semantically related entities of different ontologies. The matching process can be seen as a function f which, from a pair of ontologies to match o and o', an input alignment A, a set of parameters p e.g., weights,

Received November 22, 2012; Revised May 10, 2013; Accepted June 5, 2013

thresholds and a set of oracles and resources r, returns an alignment A' between these ontologies. This can be schematically represented in Figure 1.

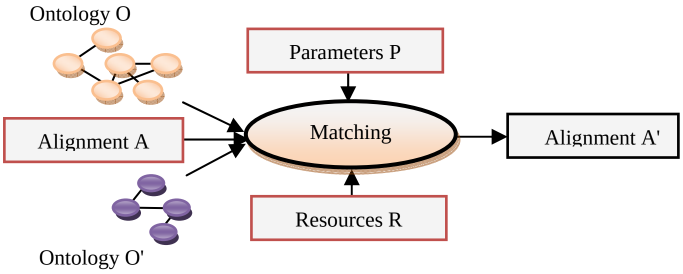

Figure 1. Matching Process

Definition 3: Correspondence

A correspondence (or mapping) between an entity e in ontology o and an entity e' in ontology o' is a 5-tuple < id;e;e';R;conf > where [8]:

a. id is a unique identifier of the correspondence.
b. e and e' are the entities (e.g. properties, classes, individuals) of o and o' respectively.
c. R is a relation such as “equivalence (=) ”, “more general (⊑) ”, “Less general (⊒)” , “disjointness (⊥)”, “overlapping (⊓) ”, holding between the entities e and e'.
d. conf is a confidence measure (typically in the [0;1] range) holding for the correspondence between
e. the entities e and e'.

Definition 4: Alignment

An alignment of ontologies o and o' is a set of correspondences between entities of o and o'. Major work of matching tools is study pairs similarities between entities and obtain the best matching between them [9], [10].

Definition 5: Constraint Satisfaction Problem (CSP)

The Constraint Satisfaction Problem is defined by:

a. a finite set of variables,
b. a function which maps every variable to a finite domain,
c. a finite set of constraints.

Each constraint limits the combination of values that a set of variables may take simultaneously. A solution of a CSP is an assignment to each variable a value from its domain satisfying all the constraints [11], [12]

2. Research Methodology

In this section we briefly referred to the generation of different similarities and refer interested readers to previous works [7] and [13] for details.

2.1. Name similarity

At first, the name similarity calculated based on the edit distance between the names of elements. The name-based similarity defined as Equation (1).

$$Namesim(\mathrm{e}_{1\mathrm{i}},\mathrm{e}_{2\mathrm{j}}) = 1 - \frac{EditDist(e_{1i},e_{2j})}{max\left(l(e_{1i},e_{2j})\right)} \quad (1)$$

2.2. Profile similarity

In order to signify each element in ontology, a profile was created. In general, the profile of a class = the class’s ID + label + comments + other constraint + its properties’ profiles + its instances’ profiles. The profile of a property = the property’s ID + label + its domain + its range.

The profile of an instance = the instance's ID + label + other expressive information. Then the tf•idf weight (Equation (2)) is allocated for each profile based on the complete collection of all profiles in the ontology.

$$\boldsymbol{w = tf.idf = \frac{n_i}{\sum_k n_k} \times log_2 \frac{N}{n}} \quad (2)$$

So in this step the cosine similarity between the profiles of two elements was measured in a vector space model using Equation (3).

$$\boldsymbol{ProfileSim}(\mathbf{e_{1i}}, \mathbf{e_{2j}}) = \frac{\mathbf{V}_{\vec{\mathbf{e_{1i}}}} \cdot \mathbf{V}_{\vec{\mathbf{e_{2j}}}}}{|\mathbf{V}_{\mathbf{e_{1i}}}||\mathbf{V}_{\mathbf{e_{2j}}}|} = \frac{\sum_{\mathbf{k=1}}^{\mathbf{n}}(\mathbf{V_k e_{1i}} \times \mathbf{V_k e_{2j}})}{\sqrt{\sum(\mathbf{V_k e_{1i}})^2} \times \sqrt{\sum(\mathbf{V_k e_{2j}})^2}} \quad (3)$$

2.3. Structural Similarity

The structural similarity between two elements derived from their structural features using Equation (4).

$$\boldsymbol{StructSim(e_{1i}, e_{2j}) = \frac{\sum_{k=1}^{n}\left(1 - diff_k(e_{1i}, e_{2j})\right)}{n}} \quad (4)$$

2.4. Similarity Aggregation

In this step, a new weight assignment method was used to adaptively aggregate different similarities that weights can set manually or with applying machine learning techniques.

2.5 Semantic Verification

The aim of this phase is to remove all kinds of inconsistency in the alignments. The output of aggregate different similarities is a similarity matrix. In this phase, the matrix values were modified. In field of ontology matching, each node signifies a hypothesis that entity e_{1i} in the first ontology can be matched with an entity e_{1j} in the second ontology and the connection between two nodes corresponds to constraint between their hypotheses. Each connection is related to a weight.

2.6 Extract mapping

Final matrix of semantic verification phase is a table that one by one mappings should be extracted from it. After aggregation, a popular greedy strategy was applied. This method is stepwise and in each step a mapping is produced, then the most similar pair is selected and their members are removed from the table. The algorithm stops when no other couple with greater similarity than a threshold is remained.

2.7. The Proposed Method

In the matching process, the size of real-world data, contain of very large resource with thousands of concepts and attributes, are extremely high. The existing techniques are mostly based on calculating similarities between entities of two ontologies by utilizing various types of information in ontologies, e.g., entity names, taxonomy structures, constraints, and entities' instances. These methods can be classified into two groups; using a single method against combining multiple methods. In the prior, all available information are defined as features in a single similarity function; while in the second, different similarity functions are defined based on different types of information, and a composite way is used to merge the results of different similarities [2]. All the matchers return a set of final alignment as matching output. Some types of inconsistencies that exist in their results are [3]:

a. Multiple-entity correspondences (Figure 2a)
b. Crisscross correspondences (Figure 2b)
c. Disjointness-subsumption contradiction (Figure 2c)
d. Subsumption and equivalence incompleteness (Figure 2d)
e. Domain and range incompleteness (Figure 2e)

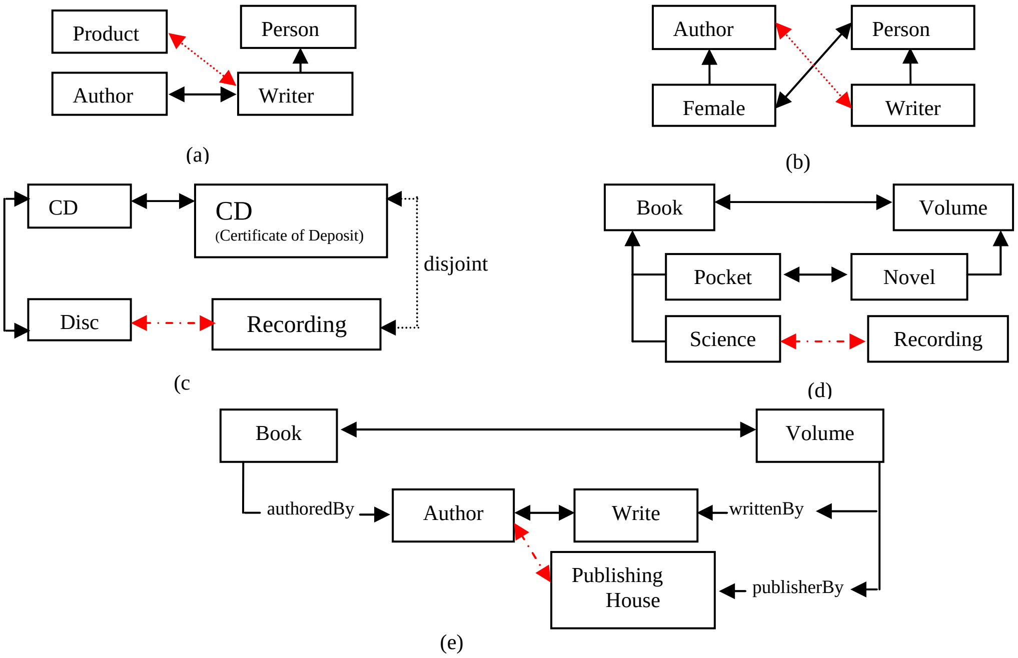

Figure 2. Kinds of Inconsistencies in Matching Alignments

Despite most matchers use different measures for finding the mappings [2], [3], [6-8], some semantic inconsistencies in final alignment are inevitable [3]. Therefore it is necessary to add a post-processing step to study error patterns in the final alignment. So it is necessary to define the number of logical constraints in the ontology with proper weights and apply one of the methods to the output alignment.

CSP arises as an intriguing research problem in ontology mapping due to the fact that the characteristics of ontology and its representations result in many kinds of constraints. To improve the quality of ontology mapping, it is required to discover a configuration that can best satisfy those constraints. CSP has already been applied at various fields [2], [3], [5], [9], [11] but its effects are rarely studied in the context of ontology matching [12]. Figure 3 shows the extraction process of the optimal alignments in the proposed method.

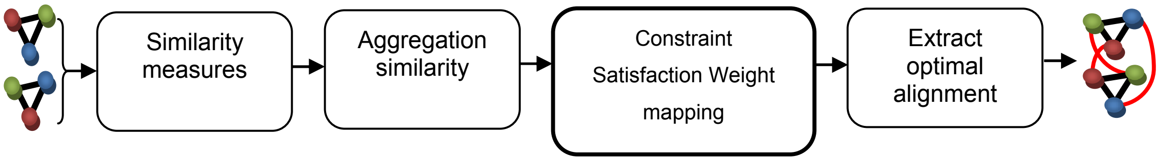
Similarity measures
Aggregation similarity
Constraint Satisfaction Weight mapping
Extract optimal alignment

Figure 3. The Architecture of the Proposed Method to Extract Optimal Alignments (Step 3)

Firstly, to compute the similarity between two ontology entities, both lexical and structural similarity measures were used. All applied constraints in some studies [2], [3], [5], [7] have the same weights which should be modified with regard to the nature of the ontology components and types of relationships between them. Therefore the proposed constraints and their weights of the following three groups are presented.

Group I: Inconsistent constraints with any degree of confidence achieved by matching tools must be removed.

a) *Constraints with weight -1*: Only 1-1 mapping is acceptable (Figure 4a).
b) *Constraints with weight -1*: If e_{1i} match e_{2k} and $e_{1j} \sqsubseteq e_{1i}$ and $e_{2k} \sqsubseteq e_{2L}$ then match e_{1j} to e_{2L} is not allowed (No crisscross mapping is acceptable) (Figure 4b).

c) *Constraints with weight -1*: Two elements match then their owl: disjointClass elements match (Figure 4c).
d) *Constraints with weight -1:* If e_{1i} match e_{2k} and $e_{1j} \sqsubseteq e_{1i}$ and $e_{2k} \sqsubseteq e_{2L}$ then match e_{1i} to e_{2L} is not allowed (Figure 4d).

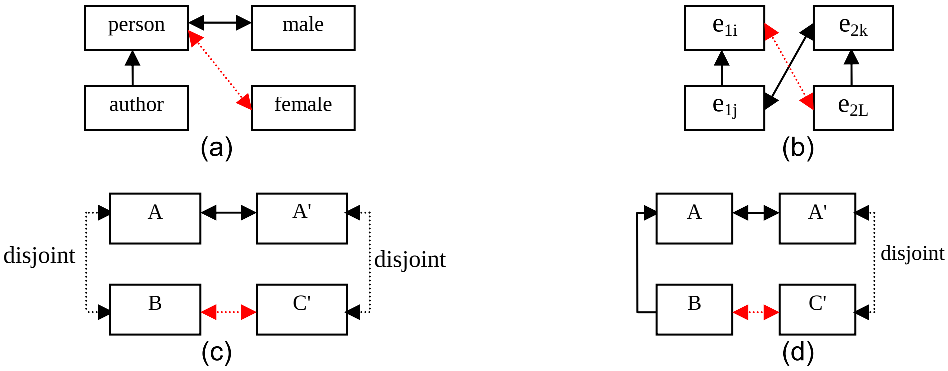

Figure 4. Constraints in Group I with Weight (-1)

Group II: certain constraints with any degree of confidence achieved by matching tools are proper, reasonable and should go out.
a) *Constraints with weight +1*: Two elements match if their owl:sameAs or owl:equivalentClass or owl:equivalentproperty elements match.

Group III: variable semantic constraints which in constraints satisfaction step, their logical and correct weights calculated and based on the concordance the decision is made. Constraints in group III with variable weight is shown in Figure 5.
a) *constraints with weight (1 / Average(number of children of two corresponding patterns))*
b) If parent elements match, then their children elements with this probability match.
c) *constraints with weight X% = (number of N1 +number of N2) / Max (N1 + N2)*
d) If *X%* numbers of children elements match, then their parent elements match. With increasing matching numbers of children also more chance of matching the two parents.
e) *constraints with weight (1 / Average(number of sibling of two corresponding elements))*
f) If e_{1i} match e_{2j} and e_{2j} has N number siblings and e_{1s} and e_{1i} are siblings in ontologies, then e_{1s} with possibility of *1 / N* match with siblings of e_{2j}. Also more math sibling of children can increase the chance of match their parents.
g) *constraints with weight (1 / Average(number of individuals of two corresponding elements))*
h) If class elements match, then their individual elements with this weight match.
i) *constraints with weight (1 / Average(number of individuals of two corresponding elements))*

If individual elements match, then their mother-class elements with this weight match.

3. Results and Analysis

3.1. Data Sets

The OAEI benchmark tests include 1 reference ontology O_R, devoted to the very narrow domain of bibliography, and multiple test ontologies, O_T, that remove different information from the reference ontology were used to evaluate how algorithms perform when information is lacking, except real cases. benchmark tests can be divided into 5 sets as shown in Table 1.

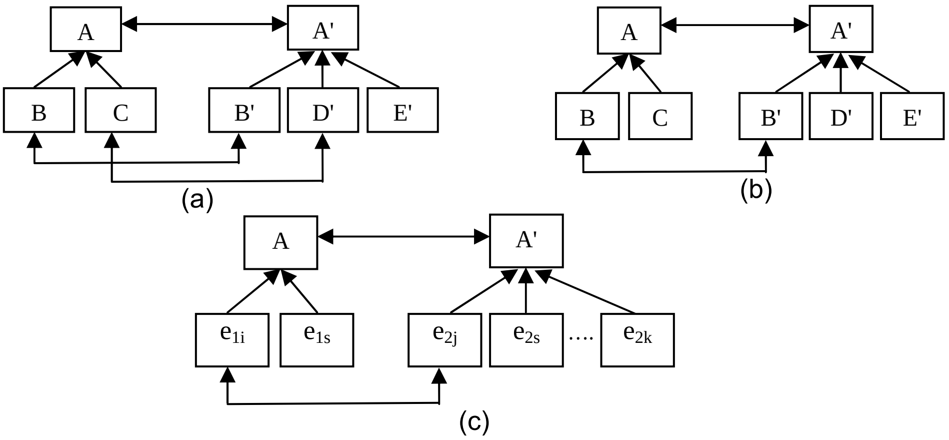

Figure 5. Constraints in Group III with Variable Weight

Table 1. The Overview of OAEI Benchmark Tests; O_R and O_T are Ontology Reference and Ontology Test Respectively

Number of ontologies	Description	Tests	Data Sets
4	O_R and O_T have exactly the same or totally different names	#101-104	D1
10	O_R and O_T have the same structure but different linguistics in some level	#201-210	D2
18	O_R and O_T have the same linguistics but different structure	#221-247	D3
15	Both structure and linguistics are different between O_R and O_T	#248-266	D4
4	O_T are real world cases	#301-304	D5

The reasons, why the OAEI benchmark tests #248-#266 were selected, are:

a. The OAEI benchmark tests is reliable tests in the area of ontology mapping.
b. The ground truth of the benchmark tests is open and hence can be used for comprehensive evaluation.
c. Tests #248-#266 are the most difficult tests among all benchmark tests. The results from all matchers on these tests are pretty lower than their results on other benchmark tests.

Therefore the improvement on these tests can greatly contribute to the overall performance of all kinds of ontology mapping approaches.

3.2. Evaluation Criteria

The main standard criteria for the evaluation of matching methods are *Precision*, *Recall* and *F-measure* that used in information retrieval. The precision, recall and f-measure are defined as (Equation (5-7)) . *Recall* and *Precision* are based on comparison alignment A to a reference alignment R and proportional similarities are discovered and undiscovered [4].

$$precision \quad p = \frac{\#correct\_found\_mappings}{\#all\_found\_mappings} \qquad (5)$$

$$recall \quad r = \frac{\#\ correct\ \_found\ \_mappings}{\#\ all\ \_possible\ \_mappings} \qquad (6)$$

$$f\text{-}measure \quad f = \frac{2 \times p \times r}{p + r} \qquad (7)$$

3.2. Experimental Design and Results

To test the proposed method, input ontologies were processed with Jena and their system designed with neural network and implemented with java. Similarity measures between ontologies computed and the similarity matrix derived. Then in the post-processing phase, the ontology semantic constraints were applied to the optimal alignment obtained based on defined weights. Proposed constraints were applied over the data set D4 (the most difficult test set). Changes in *Precision*, *Recall* and *F-measure* on each test sample of series are presented in Figure 6.

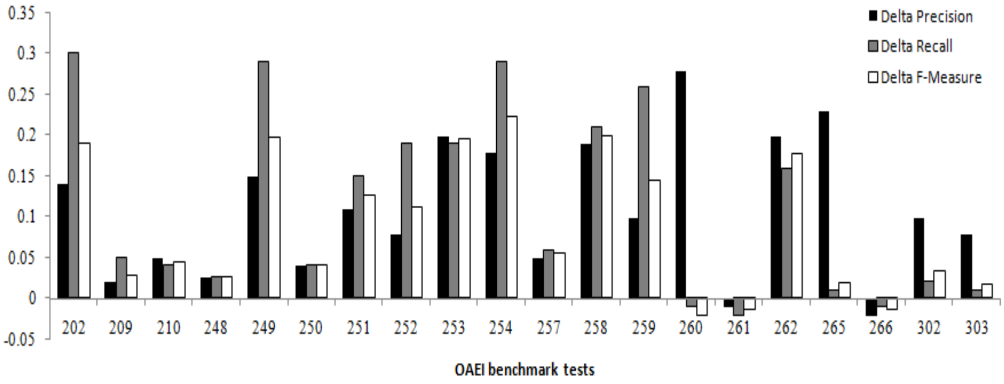

Figure 6. The Results of the Neural Network on the Date set D4

Data set D4 is the most difficult data set among all benchmark tests. As shown in Figure 6, the neural network based constraint satisfaction improved the *F-measure* of 12 tests among 15 tests except #260, #261 and #266. The largest improvement of *F-measure* happened on #254 and #262. The decrease on #261 is due to the extension of its structure, i.e., new classes are added as new layers in test ontology, which makes some constraints in neural network are not correct anymore. Meanwhile, no linguistic information was available in #261 at all, and thus there was no linguistic analysis to rely on. As shown in Figure 4, the neural network based constraints satisfaction improved 23%, 36%, and 295% for *Precision*, *Recall*, and *F-measure* respectively. Figure 7 compares the performance of proposed method and 3 top-ranked ontology mapping systems (i.e. ASMOV, PRIOR+ and Lily) on the benchmark tests at OAEI campaign 2009.

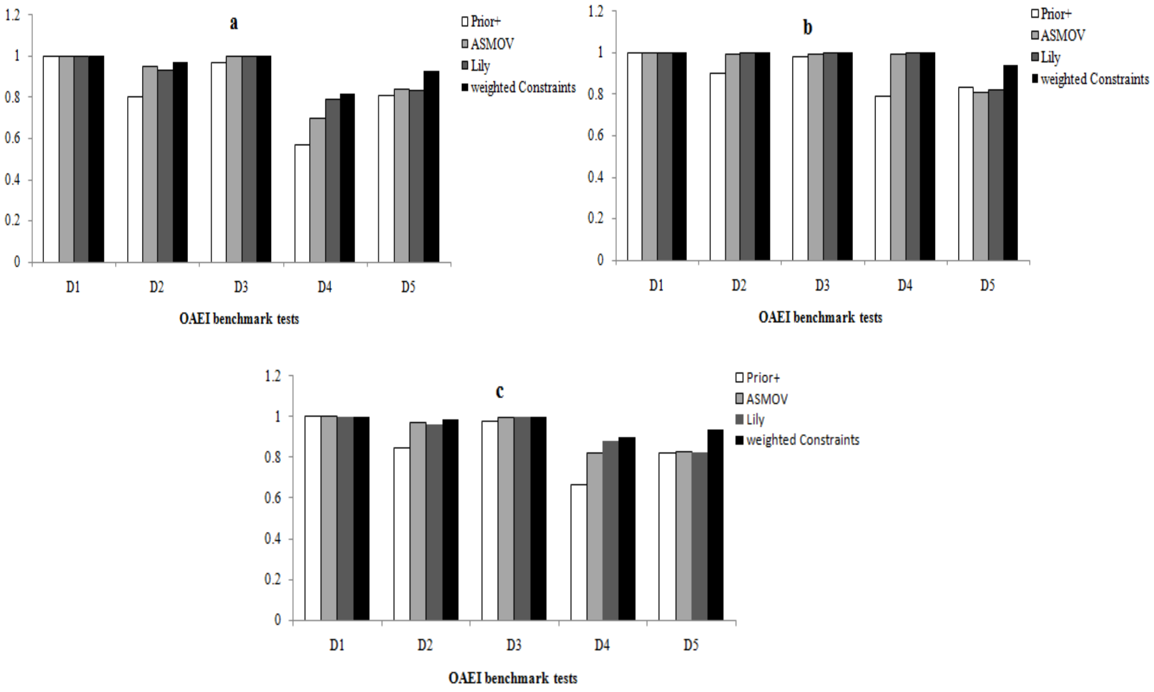

Figure 7. The Comparison of Proposed Method with Top-ranked Systems on Benchmark Tests in OAEI Campaign 2009 Based on Precision (a), recall (b) , and f-measure (c)

4. Conclusion

Ontology matching is a critical problem in many database application domains, such as data integration, E-business, data warehousing, and semantic query processing. Ontology matching algorithms play an important role in almost all phases of Semantic Web ontology engineering work such as Ontology merging, ontology alignment, ontology mapping and Ontology transformation.

In this research first different similarity measures on two sets of entrance exam OAEI ontology were calculated. Then in post-processing step, applied logical constraints in the ontology with proper weight is defined for finding the alignments to satisfy the best way to these constraints. Evaluation results showed that the standard evaluation measures are improved in the proposed approach.

References

[1] Madhavan J, Bernstein PA, Domingos P, Halevy AY. *Representing and Reasoning about Mappings between Domain Models*. 18th National Conference on Artificial Intelligence. 2002.

[2] Li J, Tang J, Li Y, Luo Q. *RiMOM: A Dynamic Multistrategy Ontology Alignment Framework*. IEEE Transactions on Knowledge and Data Engineering. 2009; 21(8).

[3] Jean Marya YR, Shironoshitaa EP, MR Kabuka. *Ontology matching with semantic verification*" Web Semantics: Science, Services and Agents on the World Wide Web. 2009; 7: 235-251.

[4] Euzenat J, Shvaiko P. *Ontology matching*. Springer. 2007.

[5] Doan A, Madhavan J, Dhamankar R, Domingos P, Halevy A. *Learning to match ontologies on the Semantic Web*. The VLDB Journal- The International Journal on Very Large Data Bases. 2003; 12: 303–319.

[6] Wang P, Xu B. *Lily: Ontology Alignment Results for OAEI 2009*" The Fourth International Workshop on Ontology Matching collocated with 8th International Semantic Web Conference USA. 2009.

[7] Mao M, Peng Y, Spring M. An *adaptive ontology mapping approach with neural network based constraint satisfaction*. Journal of Web semantics. 2009.

[8] Eckert K, Meilicke C, Stuckenschmidt H. *Improving Ontology Matching Using Meta-level Learning*" Springer-Verlag Berlin Heidelberg. 2009.

[9] Kumar V. *Algorithms for Constraint- Satisfaction Problems: A Survey.* AI MAGAZINE. 1992.

[10] Bartak R. *Constraint Propagation AND Backtracking-Based Search*. A brief introduction to ainstream techniques of constraint satisfaction. 1995.

[11] Tsang E. *Foundations of Constraint Satisfaction*. 1993.

[12] Lambrix P, Stromback L, Tan H. *Chapter 8: Information Integration in Bioinformatics with Ontologies and Standards*. SemanticTechniquesfortheWeb, LNCS5500. 2009; 343–376.

[13] Meilicke C, Stuckenschmidt H. *Analyzing Mapping Extraction Approaches*. Ontology Matching Workshop. 2007.