International Journal of Information Processing, 7(2), 1-10, 2013
ISSN : 0973-8215
IK International Publishing House Pvt. Ltd., New Delhi, India

SR-Match: Discovering Complex Semantic Matches based on Semantic Relationship of Attributes

Sonia Khetarpaul [a], S K Gupta [a]

[a]Department of Computer Science and Engineering, Indian Institute of Technology, Delhi, India,
Contact: kpaul.sonia@gmail.com

In data integration, schema matching plays an important role. Present schema matching tools combine various match algorithms, each employing a specific technique to improve matching accuracy. However there is still no fully automatic tool is available and also there is lack of accuracy. As a step in this direction, we propose a new and efficient Semantic-Relationship schema matching (SR-Match) approach which considers the semantic relationships as one of the parameters for matching. Here in SR-Match, the initial mappings performed by the basic schema mapping techniques, acts as input to the relationship matcher. Relationship matcher compares the remaining unmapped elements based on their semantic relationship with their parents and discovers remaining complex matches among attributes of two schemas. It is observed that, if both semantics and relationships are taken into account, the degree of accuracy in matching results is improved.

Keywords: Data Integration, Ontology, Schema Mapping, Semantic Relationship

1. INTRODUCTION

Continued high interest in information integration has led to a large body of research work on schema matching (E.g.,, [1-3]). Schema matching work focuses on taking as input two or more data sources (E.g., XML, OWL, Schemas) and some auxiliary information like a dictionary (E.g., WordNet) to produce a mapping between the similar elements of input data sources [1]. Schema matching for small scale applications, where the number of schema elements are less, can be done manually. But applications that require matching a large and different sets of schemes per user query, cannot have the opportunity of end user guidance. Applications where adhoc or on-the-fly data integration is needed, requires automatic schema matching approaches [4].

In today's era, user view of the database is different from the real world view. Our attempt is to map the user view into the real world accurately. In this environment users are allowed to ask any query using any website in an ad-hoc manner and a fully autonomous schema matching is not only prudent, its a necessity. To illustrate the issue on a more intuitive ground, let us consider the following example shown in Figures 1(a) and 1(b) extracted from two popular flight search web sites.

To find a good fare from Delhi to Goa, a user must search both sites manually and integrate the two schemas in his mind where he reasons that the terms depart and origin, return and desitination are similar. It is required to actually resolve such schema heterogeneity to operate efficiently. On-the-fly or ad hoc integration systems require this ability to support a platform where no such autonomous schema matching will be needed. But supporting such a flexible environment is by no means easy because current schema matchers are not equipped to handle such visions. The numerous research efforts have been done on schema matching. However automatically generating semantic mappings is a task not accomplished till date. Semi-automatic semantic mapping has gained tremendous interest in the field of databases (schema matching) [1, 2, 4-11]. Schema-based match algorithms take account of schema information and ignore instance data. We briefly examine three of im-

Algorithm 2: OntologyBasedMatcher()

Data: OntologyBasedMatcher(): Two Ontology O_1 and O_2, mapping returned from algorithm1 M' and unmarked element and their scores.
Result: Final Mapping M among attributes of two different schemas.
/*Method: In this Algorithm the output of the Algorithm 1 and two ontology are taken as input. First algorithm extract relationships specified by ontology (using function extractRelation()) between each unmarked attribute and its parents. Then we find the best mapping for each unmapped attribute u_i in S_1 with unmapped attribute v_k in S_2 based on their relationships and previous scored returned by Algorithm 1.*/

```
begin
    /* Extract Relationship of unmarked attributes (u_i of S_1, v_k of S_2)with their parent*/
    R_{u_i} ← extractRelation(u_i, P_{u_i}) ∀u_i ∈ S_1
    R_{v_k} ← extractRelation(v_k, P_{v_k}) ∀v_i ∈ S_2
    Maxscore[u_i] ← 0
    bestMatch[u_i] ← NULL
    foreach r_{u_i} ∈ R_{u_i} do
        foreach r_{v_k} ∈ R_{v_k} do
            score[u_i, v_k] ← compare(r_{u_i}, r_{v_k}) + Prevscore(u_i, v_k)
            maxscore[u_i] ← max(score[u_i, v_k], maxscore[u_i])
            if maxscore[u_i] == score[u_i, v_k] then
                bestmatch ← v_k
    Output the mapping of u_i with bestmatch
    /*ExtractRelation(u, P) function returns the parent child relationship between u and P in ontological hierarchical tree */
    Return(M)
    /* M is output mapping */
```

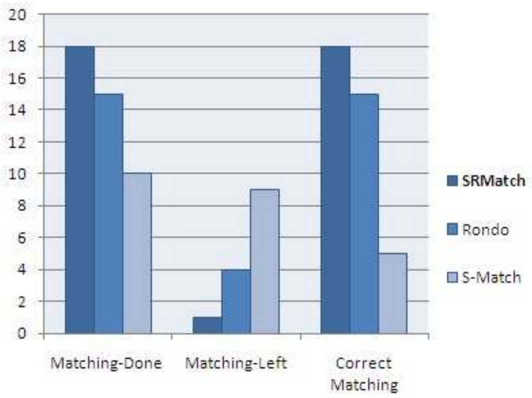

Figure 6. Comparative Analysis of SR-Match, Rondo, S-Match for Given Example

ber of correct mappings between attributes is 20. We observed that if we compare the relationships of an element with its parent along with syntax and the semantics, then accuracy of correct matching can be increased.

7. CONCLUSIONS

In this paper we proposed a schema matcher (SR-Match) which uses semantic relationships of elements as an additional parameter for schema matching. An example is simulated using SR-Match and results are analyzed and compared which shows improvement in accuracy of matching. Our main contributions include (1) designed an architecture and algorithms that gives a way for semi-automatic schema mappings based on semantic relationships of elements with their parents,(2) ordereding schema based matchers in way to improve mapping results, (3) This improved the efficiency of matcher by applying relationship matcher only on partial unmapped elements.

REFERENCES

1. H H Do and E Rahm. COMA - A System for Flexible Combination of Schema Matching Approach, *In Proceedings of VLDB*, 2002.
2. J Madhavan, P Bernstein and E Rahm. Generic Schema Matching with Cupid, *In Proceedings of VLDB*, pages 49-58, 2001.

3. R Dhamankar, Y Lee, A Doan, A Halevy and P Domingos. iMAP: Discovering Complex Semantic Matches between Database Schemas, *In Proceedings of SIGMOD*, pages 383-394, 2004.
4. A Bhattacharjee and H M Jamil. OntoMatch: A Monotonically Improving Schema Matching System for Autonomous Data Integration, *In Proceedings of the IEEE International Conference on Information Reuse and Integration*, pages 318-323, 2009.
5. D Aumller, H H Do, S Massmann and E Rahm. Schema and Ontology Matching with COMA++ (Software Demonstration), *In Proceedings ACM SIGMOD*, 2005.
6. J Euzenat and P Valtchev. Similarity-based Ontology Alignment in OWL-lite, *In Proceedings of ECAI*, pages 333-337, 2004
7. S Melnik, H Garcia-Molina, and E Rahm. Similarity Flooding: A Versatile Graph Matching Algorithm. *In Proceedings of ICDE*, pages 117-128, 2002.
8. M K Smith, C Welty and D L McGuiness. OWL Web Ontology Language Guide, Technical report, *World Wide Web Consortium (W3C)*, 2004.
9. S Melnik, E Rahm and P Bernstein. Rondo: A Programming Platform for Generic Model Management. *In Proceedings of SIGMOD*, pages 193-204, 2003.
10. P Shvaiko and J Euzenat. A Survey of Schema-based Matching Approaches, *Journal on Data Semantics (JoDS)*, IV, LNCS 3730, pages 146-171, 2005.
11. T Wang and R Pottinger. SeMap: A Generic Mapping Construction System, *In proceedings of EDBT, ACM*, Pages 97-108, 2008.
12. E Rahm, P A Bernstein. On Matching Schemas Automatically. *MSR Tech. Report*, MSR-TR-2001-17, 2001 .
13. Shoujian Yu, Zhongming Han and Jiajin Le. A Flexible and Composite Schema Matching Algorithm, *Lecture Notes in Computer Science*, Volume 3290/2004, pages 55-65, 2004.
14. F Giunchiglia, P Shvaiko and M Yatskevich. Semantic Schema Matching, *Technical Report*, DIT-05-014, University of Trento, Povo, Trento, Italy, 2005.

S K Gupta has completed his Ph.D in Graph Theory from IIT Delhi and joined as faculty in the Department of Computer Science, IIT Delhi. He is currently professor there. His academic interests include Databases, Data Mining and Graph Theory. He has guided several students for their dissertations in these areas and published extensively. He has been on the editorial board of three journals. Prof. Gupta has organised several prestigious conferences as General chair and PC chair. He has also handled several sponsored and consultancy projects. He is also in the governing board of several institutions. Having a distinguished academic career, Prof. Gupta has been a recipient of prestigious scholarship including the National Science Talent Search.

Sonia Khetarpaul is a full time Ph.D Scholar in the Department of Computer Science, IIT Delhi. She completed her M.Tech (IT) from GGSIP University and has worked as Assitant Professor for 9 years in the colleges affiliated with GGSIP university. Her academic interests include Databases, Data Integration and Spatio-temporal Data Mining.