Comparing Similarity Combination Methods for Schema Matching

Eric Peukert

SAP AG, SAP Research
Chemnitzer Str. 48
01187 Dresden, Germany
eric.peukert@sap.com

Sabine Maßmann, Kathleen König

WDI-Lab - Institut für Informatik
Universität Leipzig
04009 Leipzig, Germany
{massmann, koenig}@informatik.uni-leipzig.de

Abstract: A recurring manual task in data integration or ontology alignment is finding mappings between complex schemas. In order to reduce the manual effort, many matching algorithms for semi-automatically computing mappings were introduced. In the last decade it turned out that a combination of matching algorithms often improves mapping quality. Many possible combination methods can be found in literature, each promising good result quality for a specific domain of schemas. We introduce the rationale of each strategy shortly. Then we evaluate the most commonly used methods on a number of mapping tasks and try to find the most robust strategy that behaves well across all given tasks.

1 Introduction

Finding mappings between complex schemas is crucial in many areas such as data integration or ontology alignment. Due to the heterogeneity of schemas identifying such schema mappings is often a complex and time consuming process. In order to speed up that process, semi-automatic matching techniques were developed. These techniques rely on algorithms, so called matchers, to compute correspondences between elements of schemas. After computing similarities based on syntactical, linguistic and structural schema and instance information, the user is provided with the most likely mapping candidates for further refinement [RB01, SE05].

In the last decade, it turned out that a combination of the results of a number of individual matchers often improves the mapping result quality. The idea is to combine complementary strengths of different matchers for different sorts of schemas. Current systems execute a number of matchers, combine their results and finally select the most promising element pairs for the final mapping. Achieving good result quality highly depends on choosing the most appropriate result combination and selection method. All proposed techniques [DR02] try to compute a single result out of a number of base matcher results. The combination approaches differ in the parameterization effort and the result quality retrieved by applying a certain combination or selection method. Here it would be desirable to know about the strategies robustness for a set of mapping tasks.

In this paper we want to evaluate a range of combination strategies on a number of mapping tasks. Our goal is to find the most suitable combination method for each mapping task as well as the most robust strategy. We define robustness as the ability of a matching strategy to return good results for different matching tasks without bigger outliers. Our results support the user in choosing the most appropriate combination strategy for different use cases. In summary, our contributions are the following:

- The paper gives an introduction of the most common combination methods and their rationale. A focus is set onto strategies that do not require additional configuration effort.
- The achieved quality of the presented methods is evaluated on a number of different mapping tasks.
- All strategies are evaluated with respect to their robustness. Our results show, that no strategy returns the best results in all mapping task. However we see some strategies being more robust than others.

2 Common Matching Process

To better understand where combination methods are needed in a matching process, an overview to a general matching process is described shortly.

All currently promoted matching systems use a combination of different matching techniques (see surveys in [RB01, SE05]) for improving the quality of the matching results. In our work we restrict ourselves to the most common system architecture of parallel combination that was first introduced by COMA [DR02]. However other topologies can also be used to combine matching techniques like sequential combination or iterative computation [LTL09, MBR01].

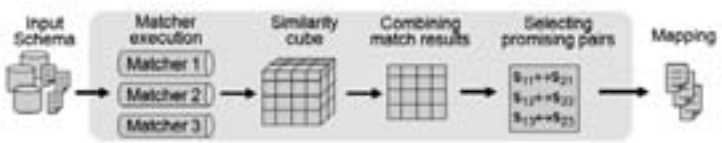

Figure 1: Common Matching Process

In parallel combination systems all matchers are executed independently, typically on the whole cross product of source- and target schema elements (see Figure 1: Common Matching Process). Results of individual matchers, so called similarity matrices, are put into a similarity cube [DR02]. A similarity combination operation reduces the cube down to a single similarity matrix. A subsequent selection operator tries to select the most promising element pairs, i.e. by using a threshold. Some systems post-process the found mapping with a constraint resolving step to prune out conflicting mappings. In this paper we particularly focus on the step that reduces the similarity cube to a single similarity matrix (Combining match results), what we call similarity combination.

3 Combination Techniques

The most commonly used methods for similarity combination are MIN, MAX, AVERAGE and WEIGHTED[DR02]. Recently also HADAPT[MYM08], SIGMOID[ES04], OWA[JHQ08], OPENII[Sel10] and NONLINEAR[Alg10] were proposed. In the following paragraphs each of these combination approaches will be described shortly.

3.1 MIN/MAX Combination

The MIN-Combination always chooses the minimum value of a set of values that were computed by different matchers. This approach is very pessimistic, since it requires all matchers to return high similarity values to later "survive" a selection. The MAX-combination in comparison behaves very optimistically since only one matcher needs to return a high similarity value, no matter what other matchers compute.

3.2 Weighted Sum Approaches

According to [ES04] similarity combination combines multiple similarity values from k matchers to one value. It can be expressed through an adjusted weighted sum of input similarities:

$$\mathrm{sim}_{\mathrm{agg}}(\mathrm{s},\mathrm{t}) = \frac{\sum_{\mathrm{k}=1\ldots\mathrm{n}} \mathrm{w_k} \cdot adj_{\mathrm{k}}(\mathrm{sim_k}(\mathrm{s},\mathrm{t}))}{\sum_{\mathrm{k}=1\ldots\mathrm{n}} \mathrm{w_k}} \qquad (3.1)$$

with (s,t) being a source and target element pair, $sim_k(s,t)$ being the similarity computed by the k-th matcher, $\mathrm{w_k}$ being a weight for each individual matcher and function $adj: [0,1] \rightarrow [0,1]$ being an adjustment function to transform the original similarity. The adjustment function is a continuous and not necessarily differentiable function. Most of the combination approaches we describe represent a special case of this adjusted weighted sum of input similarities. Some techniques focus on how the weights are defined like OWA, OPENII and HADAPT and others focus on the adjustment function like SIGMOID does.

There are a number of techniques known in literature that apply machine learning techniques for finding the best weights for a given mapping problem or problem class [ES04, ES05, MG08]. However, these learning-based approaches are not in the focus of this paper. The reason is that gold standard mappings are rare, thus making learning approaches often impossible to use. Also we do not consider approaches that support the user in manually defining the weights proposed in the CMC-Method [TY05].

The **AVERAGE combination** is the simplest version of the weighted approaches. It assumes equal weights for every matcher and uses the identity function as adjustment function. AVERAGE showed good results on former evaluations [Do05] since it levels out the individual weaknesses and strength of individual matchers

The **WEIGHTED combination** also uses the identity function as adjustment function. The simplified equation computes a weighted sum of matcher input similarities. WEIGHTED crucially depends on the optimal setting of weights for each matcher.

The **HADAPT combination** method presented in the PRIOR+ System automatically determines the weights for a weighted combination. It relies on a measure that is called harmony (Note: Their naming clashes with the Harmony system[Mo08]), that is computed from the output similarities of individual matchers. The overall idea is to count entries in the similarity matrix that both are a maximum in a line and a row of the matrix. Their measure is related to the Direction-Both Selection method introduced in COMA [RB01] and the Stable Marriage Property [GI89]. The main assumption is to give match results with higher harmony-value a higher weight since those values might have computed better results. In their work a correlation of harmony with the F-Measure was shown so that harmony could be an indicator for a good F-Measure.

The **OWA combination** (Ordered Weighted Average) tries to simplify the process of determining the weights of individual matchers. For that purpose, each set of similarity values computed for an element pair is ordered and each position in the ordered list gets a weight assigned. It implies that different matcher-similarities might have a different position and weight for two different element comparisons. Additionally the authors of OWA proposed a so called linguistic method to come up with combination weights automatically. Their linguistic approach proposes a number of variants like OWAMOST or OWAALH and also Maximal and Minimal which are equal to MIN, MAX.

The **SIGMOID combination** prepares the matcher results for the weighted sum by setting the adjustment function to:

$$adj(x) = \frac{1}{1+e^{-t(x-s)}} \tag{3.2}$$

where t sets the slope and the s describes a shifting factor for the sigmoid function. These values can be adjusted to a given mapping task. It pre-processes the input match results by increasing higher similarity values and decreasing lower values. It acts similar to a contrast filter in image processing by increasing the contrast on input matrices. The sigmoid function can also be described as a smoothed threshold by interpreting a threshold as a stepping function.

The **OPENII combination** method gives higher weight to higher similarity values and lower weight to lower similarity values. In that respect it is similar to the SIGMOID approach. The difference is that they directly use the absolute value of their so called voter score as a weight to compute a confidence score. Voter scores are similar to similarity values except that voter scores are in the interval $[-1,1]$. Values higher than 0 get a high confidence, whereas values below 0 get a low confidence:

$$conf(\mathrm{s},\mathrm{t}) = \frac{\sum_{\mathrm{k}=1\ldots\mathrm{n}} |\mathrm{ms_k}(\mathrm{s},\mathrm{t})| \cdot \mathrm{ms_k}(\mathrm{s},\mathrm{t})}{\sum_{\mathrm{k}=1\ldots\mathrm{n}} |\mathrm{ms_k}(\mathrm{s},\mathrm{t})|} \tag{3.3}$$

Since all other combination methods rely on similarities in the interval $[0,1]$ we introduce functions $\mathrm{t}_1 : [0,1] \rightarrow [-1,1]$ and $\mathrm{t}_2 : [-1,1] \rightarrow [0,1]$ that transform

similarities from and to voter scores. By using these functions we can adapt the OPENII approach for our evaluation.

$$\mathrm{sim}_{\mathrm{agg}}(s,t) = t_2\left(\frac{\sum_{k=1\ldots n} |t_1(\mathrm{sim}_k(s,t))| \cdot t_1(\mathrm{sim}_k(s,t))}{\sum_{k=1\ldots n} |t_1(\mathrm{sim}_k(s,t))|}\right) \quad (3.4)$$

The **NONLINEAR combination** also relies on weights but follows an extended approach. It tries to include interdependencies of similarity measures into the combination of their similarities for different matchers:

$$\mathrm{sim}_{\mathrm{agg}}(s,t) = \lambda \sum_{k=1}^{N} w_k \mathrm{sim}_k(s,t) \pm (1-\lambda) \sum_{j=1}^{N} \sum_{k=j}^{N} \mathrm{sim}_j(s,t)\mathrm{sim}_k(s,t) \quad (3.5)$$

The first part of the formula computes the weighted average similar to equation 3.1 except that the weights are not normalized on the sum of weights. The second part is computing the correlations between similarity measures. Depending on the value of the weighted average, the value will be added or subtracted which is implied by the $\pm$. This value behaves similar to the shifting factor of the SIGMOID combination and is hard to set. The constant λ is used to level computed values into the interval $[0,1]$.

4 Comparative Evaluation

In our evaluation we first characterize our data set that consists of a number of real world mapping tasks. We then describe our evaluation methodology where we tried to fix some variables to simplify comparison of combination methods. After that our evaluations are presented and analyzed.

4.1 Datasets

Table 1: Evaluation Data Set

Mapping Task	Dimension	#C	Resolution
CIDX_Apertum	40x147	54	Paths
CIDX_Excel	40x54	65	Paths
CIDX_Noris	40x65	32	Paths
CIDX_Paragon	40x80	49	Paths
Excel_Apertum	54x147	79	Paths
Excel_Noris	54x65	50	Paths
Excel_Paragon	54x80	60	Paths
Noris_Apertum	65x147	85	Paths
Noris_Paragon	65x80	45	Paths
Paragon_Apertum	80x147	66	Paths
DB_Mapping	19x20	11	Paths
s3Mapping	125x123	67	Paths

Mapping Task	Dimension	#C	Resolution
dmoz_google	746x728	729	Paths
dmoz_web	746x418	218	Paths
dmoz_yahoo	746x1132	356	Paths
Freizeit	71x67	67	Paths
google_web	728x418	211	Paths
google_yahoo	728x1132	340	Paths
Lebensmittel	59x53	32	Paths
web_yahoo	418x1132	197	Paths
OAEI_101-301	80x55	54	Nodes
OAEI_101-302	80x42	43	Nodes
OAEI_101-303	80x126	43	Nodes
OAEI_101-304	80x74	64	Nodes

In our evaluations we use four groups of data sets (see Table 1). #C represents the number of intended correspondences.

- A number of mappings between schemas of the purchase order domain are taken from the COMA++ Evaluation [Do05] (CIDX, Apertum, Excel, Noris, Paragon). These schemas exhibit recurring features of business schemata such as a strong reuse of components, camel-case naming and different data types.
- The second group consists of mappings between schemas from the Spicy-Evaluations [Bon08] that are database schemata with foreign key relationships.
- The third group is taken from the domain of web directories (dmoz, Google, Yahoo, web). These schemas are taxonomies with deep paths and nodes without types.
- The last group consists of four mappings from the recent OAEI Ontology alignment contest [Eu09]. Here we restricted the set to real world alignments since the other reference alignments are synthetically generated gold standards.

In literature other Benchmarks for schema matching systems were proposed such as XBenchMatch[DBH07] and STBenchmark[ATV08]. XBenchMatch consists of only four small sized mapping problems. They were left out of the collection since it already consisted of number of other small mapping problems. STBenchmark generates synthetic schemata and mappings. Since we restricted our selection of mapping task to real world examples the STBenchmark was not included.

4.2 Experimental Methodology

For our evaluations we implemented all strategies from Section 3 that are: MIN, MAX, AVERAGE, WEIGHTED, HADAPT, SIGMOID, OWAMOST, OPENII and NON-LINEAR. Since WEIGHTED requires a manual definition of weights we evaluate that strategy separately. All other strategies are used with a fixed parameter setting proposed by the original authors on all mapping tasks. For SIGMOID we take the values applied by the NOM-System [ES04] that is $t = 8$ and $s = 0.5$. For NONLINEAR we chose $\lambda = 0.5$ and subtract the second part of the NONLINEAR- equation for values lower than 0.3 in the first part. As evaluation measure we apply the commonly used Precision and Recall as well as the F-Measure that combines both.

In order to reduce the search space we first tried to find an optimal parameterization for the selection step that takes place after the combination. From recent evaluations [Do05] we took the meta-data based COMA_OPT matcher that consists of 4 matchers (Name, Path, Leaves and Parents). Given these matchers we computed the F-Measure on our given dataset for all combination strategies and different selection techniques. According to [DR02] a number of selection techniques can be used that are DIRECTION, THRESHOLD, MAXDELTA, and MAXN. As selection direction BOTH was chosen. The THRESHOLD selection was parameterized with values ranging from 0 to 1 using 0.01 steps, a MAXDELTA-selection with values from 0 to 0.6 with 0.01 steps and MAXN selection with values from 0-10 for the N. Details on these strategies can be found in [DR02].

In our observations the best selection strategy to choose is the MAXDELTA, with a delta value of 0.01. We decided to fix the selection parameter to MAXDELTA 0.01 for the upcoming evaluations of the different combination approaches.

4.3 Comparison of Combination Methods

First we compared all combination strategies on their best F-Measure for each mapping task (see Figure 2). The x-axis enumerates the different mapping tasks, whereas the y-axis shows the achieved F-Measure of a combination strategy. The mapping tasks are ordered by the best F-Measure that was achieved with at least one of the configurations (see Best FM in Figure 2). Each line represents one combination method. For reasons of readability we created two separate figures, each containing 4 strategies. Obviously the MIN and MAX combination do not perform well in comparison to the others. MAX never returns better results than all others but often worse. Surprisingly the MIN-strategy returns the best F-Measure for the Excel_Paragon and DB-Mapping mapping task.

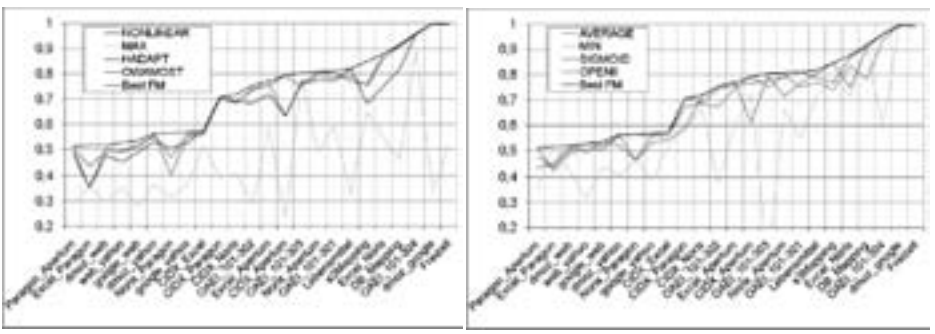

Figure 2: Maximum F-Measure for each mapping task

OWAMOST performs better than MIN/MAX and more often returns the best F-Measure possible. On the other hand it also produces negative outliers in a number of other mapping tasks. This could be explained by the rationale of OWAMOST to throw away very high and very low values. This behavior is prone to error since all matcher similarities of our 4-matcher set should be considered. OPENII performs little better than OWAMOST by having less negative outliers. Surprisingly the OPENII combination seems to have problems mainly on the purchase order mappings. This could be explained by its behavior to overweight higher similarities and underweight lower similarities. In order to distinguish different contexts of shared components small differences in the path similarities are highly relevant. If those differences are underweighted the elements get mapped into the wrong context producing bad results.

The best 4 strategies found are AVERAGE, NONLINEAR, HADAPT and SIGMOID. HADAPT is in most cases not much different from AVERAGE since the computed weights often equal to AVERAGE. HADAPT seems to have problems with some of the purchase order schemata. And again the reuse of components in the purchase order schemata and the contained 1:n mappings give the explanation. The computed harmony-value expects 1:1 correspondences in the final result. Hence, it underweights matchers

that produce 1:n correspondences. There is almost no difference between NONLINEAR and AVERAGE. Obviously the interdependencies of matcher similarities do not have a big influence in our matcher set, giving the first part of equation 3.5 the most influence which is an AVERAGE combination. SIGMOID behaves different to the others and often returns results that are slightly below the maximum. In some rare cases it performs better than all others. The problem with the SIGMOID combination is the definition of the shifting factor. High shifting values might reduce similarity values for too many element pairs that otherwise would have contributed to the final result. By coincident in some mapping tasks our parameter setting seemed to be optimal, whereas in most other tasks it only decreased the quality.

In our evaluations we decided to treat the WEIGHTED combination separately since it requires a manual definition of weights. We a number of combinations of weights and applied them on our mapping tasks. Figure 3 shows a comparison of the AVERAGE strategy with the maximal and minimal possible F-Measure with different weights for each matcher.

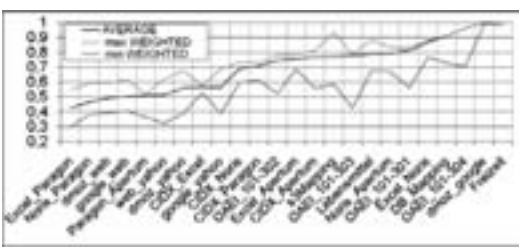

Figure 3: Comparing AVERAGE to WEIGHTED with different weights

Obviously by setting the proper weights, the F-Measure could be increased for individual mapping tasks using the WEIGHTED strategy. However setting the proper weights is difficult, even for trained experts. Interestingly by setting the wrong weights the F-Measure decreases more than it could increase by setting the right weights. This can be explained by the fact that not only strengths of matchers are weighted higher but also weaknesses might get overweighted. Thus using the WEIGHTED strategy imposes a higher risk of achieving bad mapping results. Also we did not find a combination of weights that performed significantly better than AVERAGE in all mapping tasks.

In our evaluations we took average weights for the NONLINEAR and the SIGMOID combination. However both combination methods allow specifying manual matcher weights. For that reason we also compared both, NONLINEAR and SIGMOID, to a weighted equivalent. The results did not differ much from the results in Figure 3 and we left them out for space reasons.

Initially we restricted our evaluation to a set of four matchers. However, in order to show the influence of the number of matchers, we ran our experiments on sets of 1 to 7 matchers. For each set we computed the best possible selection of matchers. Figure 4a compares the achieved average F-Measure over all mapping tasks for each combination method. The result shows that applying more than 4 matchers decreases the result quality

which was already shown by [Do05]. The best combination strategy we found with each matcher set was AVERAGE, closely followed by NON-LINEAR. Obviously all combination approaches that try to automatically define the weights or pre-adjust similarity values seem to have problems with many matchers. Also for a small set of matchers, the automatic definition of weights works. But on higher numbers of matchers the AVERAGE combination returns better results, even though taking too many matchers is not recommended since the result quality drops.

Finally we want to find robust strategies. The average result of a strategy over different mapping tasks (as shown in Figure 4) does not necessarily show its robustness. For that reason, we computed the variance of deviation from the possible maximum F-Measure (with the 4-matcher set) and visualized the result in Figure 4b. As described above, AVERAGE and NONLINEAR behave almost equal in our test cases with AVERAGE returning slightly better results. However, when looking at the variances there is a difference. NONLINEAR shows a higher variance on the OAEI-Tasks and the Spicy Tasks. We therefore conclude that AVERAGE behaves more robust in our different mapping tasks. When comparing SIGMOID and HADAPT, the effect is even stronger. Both have similar average values but HADAPT has a much higher variance in the PO and Spicy mapping tasks. Thus SIGMOID is much more robust in comparison to HADAPT.

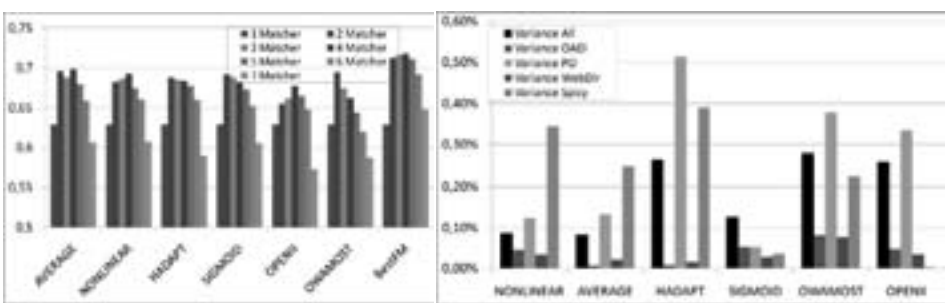

Figure 4: (a) Averaged F-Measure for three matcher sets, (b) Best variance by mapping task group

5 Conclusion and Outlook

In this paper we introduced a number of commonly used methods for combining similarities of different matchers. We evaluated each of them on a number of mapping tasks that are well known in the schema matching community.

Some of our results are surprising. There is no single strategy that is returning the best results in all test cases. In our mapping tasks we could not find an argument for using MIN/MAX strategies since they do not perform well in almost all our tasks. The AVERAGE and NONLINEAR strategy performed best in our evaluations. However the influence of interdependencies in NONLINEAR was so small that it almost computed equal to AVERAGE. HADAPT, SIGMOID, OPENII and OWAMOST tried to automatically set the weights or adjusted individual similarities. In some cases this

improved the result but in other cases its effect was negative. Also we found strategies that are more robust than others.

In future we need to find out from the mapping task, when each of these strategies should be applied. In that context, pre-processing of input schemata is crucial. Also a combination of automatically setting the weights and automatically adjusting similarity values could be promising. Our comparison of AVERAGE to WEIGHTED with a manual setting of weights showed some potential that is still to uncover in future.

References

[Alg10] Alsayed Algergawy: Management of XML Data by Means of Schema Matching. Dissertation Otto-von-Guericke-Universität Magdeburg 2010

[Bon08] Bonifati, A.; et. al.: The Spicy system: towards a notion of mapping quality. SIGMOD Proc., 2008

[ATV08] Alexe, B., Tan, W., and Velegrakis, Y. 2008. STBenchmark: towards a benchmark for mapping systems. VLDB Proc., 2008, pp. 230-244.

[Do05] Do, H.: Schema Matching and Mapping Based Data Integration. Dissertation, University of Leipzig, 2005

[DR02] Do, H. H. & Rahm, E.: COMA - A System for Flexible Combination of Schema Matching Approaches. VLDB Proc., 2002

[DBH07] Duchateau, F.; Bellahsène, Z.; Hunt, E.: XBenchMatch: a benchmark for XML schema matching tools. VLDB Proc., 2007

[ES04] Ehrig, M.; Staab, S.: QOM - Quick Ontology Mapping. ISWC, 2004, pp. 683-697.

[ES05] Ehrig, M.; Staab, S. & Sure, Y. Bootstrapping ontology alignment methods with APFEL. WWW '05, 2005, p. 1148-1149

[Eu09] Euzenat et al.: Results of the Ontology Alignment Evaluation Initiative 2009, Second International Workshop on Ontology Matching, 2009

[GI89] Gusfield, D. & Irving, R. W. The stable marriage problem: structure and algorithms MIT Press, 1989

[JHQ08] Ji, Q., Haase, P., Qi, G. Combination of Similarity Measures in Ontology Matching by OWA Operator, IPMU'08, 2008

[LTL09] Li, J.; Tang, J.; Li, Y.; Q. RiMOM: A Dynamic Multistrategy Ontology Alignment Framework. IEEE TKDE, 2009, 21, pp. 1218-1232

[MBR01] Madhavan, J.; Bernstein, P. A. ; Rahm, E.: Generic Schema Matching with Cupid. VLDB Proc., 2001.

[MG08] Marie, A.; Gal, A.: Boosting Schema Matchers. OTM Proc. 2008, pp. 283-300

[Mo08] Mork, P. et. al.: The Harmony Integration Workbench. 2008, pp65-93

[MYM08] Ming M.; Yefei P.; Michael S.: A Harmony Based Adaptive Ontology Mapping Approach. SWWS'08 , 2008

[RB01] Rahm, E.; Bernstein, P. A.: A survey of approaches to automatic schema matching. The VLDB Journal, 2001, 10, pp. 334-350.

[SE05] Shvaiko, P. & Euzenat, J.: A Survey of Schema-Based Matching Approaches. Journal on Data Semantics IV, 2005.

[Sel10] Seligman L. et. al.: OpenII: An Open Source Information Integration Toolkit, SIGMOD Proc., 2010

[TY05] Tu, K., Yu, Y.: CMC: Combining mutiple schema-matching strategies based on credibility prediction. (DASFAA), pp. 17-20, 2005, China.