# Optima Results for OAEI 2011

Uthayasanker Thayasivam and Prashant Doshi

THINC Lab, Department of Computer Science, University of Georgia, Athens, Georgia 30602
uthayasa,pdoshi@cs.uga.edu

**Abstract.** In this report, we present the results of Optima in the Ontology Alignment Evaluation Initiative (OAEI) 2011. We participate in three tracks of the campaign offered in SEALS platform: Benchmark, Conference and Anatomy. We review the iterative ontology alignment approach adopted by Optima and its results for the Benchmark and Conference tracks.

## 1 Presentation of the system

The increasing usefulness of the semantic Web is in part, due to an increase in the number of ontologies on the Web. Applications such as Web service compositions and semantic Web search, which utilizes these ontologies demand a way to align these ontologies. Nowadays numerous ontology alignment tools exist. They can be broadly identified using, 1) the level of human intervention needed; 2) the amount of prior training data needed; and 3) the facets of ontologies used and the way they are utilized. We present a fully automatic, general purpose ontology alignment tool called Optima [2], which does not need any prior training. Like many other tools, Optima utilizes both lexical and structural facets of ontologies to arrive at an alignment. However, it primarily differs in a different aspect – being iterative – from most other alignment tools that presently exists. Common approaches build an alignment in a single pass using a variety of heuristics and similarity measures. In contrast to single pass approaches Optima continues to improve an alignment in an iterative fashion. Optima formulates the problem of inferring a match between two ontologies as a maximum likelihood problem, and solves it using the technique of expectation-maximization (EM). Specifically, it adopts directed graphs as its model for ontology schemas and uses a generalized version of EM to arrive at a map between the nodes of the graphs. At the end of each iteration, Optima derives a possibly inexact match. Inexact matching is the process of finding a best possible match between the two graphs when exact matching is not possible or is computationally difficult.

We describe briefly the formal model of an ontology as utilized by Optima and the EM-based algorithm adopted by Optima in the next two subsections.

### 1.1 Ontology Model

Optima adopts the common directed labeled graph model for ontology schemas where the nodes of the graphs are the concepts (named classes in RDFS and OWL) and the labeled edges are the relationships (properties) between the classes. Contemporary languages for describing ontologies such as RDFS and OWL also allow the ontologies to be modeled as directed labeled graphs [3]. Because Optima focuses on identifying a many-one map, let the graph with the larger number of nodes be labeled as the *data* graph while the other as the *model*. Formally, the data graph is modeled as: $O_d = \langle V_d, E_d, L_d \rangle$, where $V_d$ is the set of labeled vertices representing the concepts, $E_d$ is the set of edges representing the relations which is a set of ordered two subsets of $V_d$, and $L_d : E_d \rightarrow \Delta$ where $\Delta$ is a set of labels, gives the edge labels. Analogously, $O_m = \langle V_m, E_m, L_m \rangle$ is the model graph against which the data graph is matched. Let M be the standard $|V_d| \times |V_m|$ matrix that represents the match between the two graphs:

$$M = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1|V_m|} \\ m_{21} & m_{22} & \cdots & m_{2|V_m|} \\ . & . & \cdots & . \\ . & . & \cdots & . \\ . & . & \cdots & . \\ m_{|V_d|1} & m_{|V_d|2} & \cdots & m_{|V_d||V_m|} \end{bmatrix} \tag{1}$$

Each assignment variable in $M$ is,

$$m_{a\alpha} = \begin{cases} 1 \text{ if } f(x_a) = y_\alpha : x_a \in V_d, y_\alpha \in V_m \\ 0 \text{ otherwise} \end{cases}$$

where $f(\cdot)$ represents the correspondence between the two ontology graphs. Consequently, $M$ is a binary matrix representing the match.

### 1.2 EM-based Algorithm

Optima views the mapping between two ontologies as the problem of, the concepts of source ontology (data graph) emitting the concepts of target ontology (model graph) with an underlying Bernoulli distribution. It formulates this model as a maximum likelihood problem and solves it using the popular expectation maximization algorithm (EM) developed by Dempster et al. [1] to find the maximum likelihood estimate of the alignment from observed data instances in the presence of missing correspondence. It iteratively searches for the match matrix, $M_*$ , that gives the maximum conditional probability of the data graph, $\mathcal{O}_d$, given the model graph, $\mathcal{O}_m$, and the match assignments. Formally,

$$M_* = \underset{M \in \mathcal{M}}{\arg\max}\, Pr(\mathcal{O}_d | \mathcal{O}_m, M)$$

where $\mathcal{M}$ is the set of all match matrices. While there may be as many as $2^{|V_d||V_m|}$ possible alignments, Optima shrinks this space by considering many-one maps only. In the equation above, Optima uses heuristics to guide its search space. Section 1.4 explains the heuristics used in Optima.

$$Pr\left(\mathcal{O}_d | \mathcal{O}_m, M\right) = \prod_{x_a \in V_d} \sum_{y_\alpha \in V_m} Pr(x_a | y_\alpha, M) \pi_\alpha \tag{2}$$

where $\pi_\alpha = Pr(y_\alpha|M)$ is the prior probability of the model graph vertex, $y_\alpha$, given the match matrix, $M$. The correspondence, $f$, is hidden from us. The matrix $M$ may be seen as a mixture model by viewing each assignment variable, $m_{a\alpha}$, as a model.

This modeling does not have an inherent way of finding mapping between edges. Though it is viable for Optima to map the bipartition transformation of the provided graph it avoids it for the excessive complexity involved. Hence, Optima additionally allows matching the concept graph and labeled relationships as separate but dependent tasks.

**E Step** Optima formulates a conditional expectation of the log likelihood with respect to the hidden variables given the data graph and a guess of the match matrix, $M^n$ at some iteration n, in order to find the most likely match matrix:

$$\begin{aligned} Q(M^{n+1}|M^n) &= E\left[logPr(x_a|y_\alpha, M^{n+1})\pi_\alpha^{n+1}|x_a, M^n\right] \\ &= \textstyle\sum_{a=1}^{|V_d|}\sum_{\alpha=1}^{|V_m|} Pr(y_\alpha|x_a, M^n)\; logPr(x_a|y_\alpha, M^{n+1})\pi_\alpha^{n+1} \end{aligned} \tag{3}$$

Optima derives the probability that the data graph node, $x_a$, is in correspondence with the model graph node, $y_\alpha$, under the match matrix of iteration $n$ , $M^n$ as ,

$$Pr(x_a|y_\alpha, M^n) = \left[\tfrac{1}{Pr(x_a|y_\alpha)}\right]^{|V_d||V_m|-1} \textstyle\prod_{b=1}^{|V_d|}\prod_{\beta=1}^{|V_m|} Pr(x_a|y_\alpha, m_{b\beta}^n) \tag{4}$$

Here, it is assumed that the individual models, $m_{b\beta}^n$, are independent of each other.

Optima extends the structural graph matching initially proposed by Luo and Hancock [5] with label similarity measures to derive the probability that $x_a$ is in correspondence with $y_\alpha$ given the assignment model, $m_{b\beta}$.

$$Pr\left(x_a|y_\alpha, m_{b\beta}^n\right) = (1 - P_\epsilon(x_a, y_\alpha))^{EC}\, P_\epsilon(x_a, y_\alpha)^{1-EC} \tag{5}$$

where the correspondence error, $P_\epsilon : V_d \times V_m \to [0, 1]$, is defined as,

$$P_\epsilon(x_a, y_\alpha) = P_e(|V_d|, |V_m|) - \delta \times P_s(x_a, y_\alpha) \tag{6}$$

EC denotes the edge consistency between the two graphs, which is defined as,

$$EC = \begin{cases} 1 & \langle x_a, x_b\rangle \in E_d \wedge \langle y_\alpha, y_\beta\rangle \in E_m \wedge m_{b\beta} = 1 \\ 0 & \text{otherwise} \end{cases}$$

The correspondence error, $P_\epsilon$, is based on the structural error, $P_e\ (|V_d|, |V_m|)$ , a function based on the sizes of the graphs, and the similarity of the node labels, $P_s(x_a, y_\alpha)$. Parameter $\delta \in [0, 1]$ controls how much weight is given to the similarity between entity labels. The structural error is defined as,

$$P_e(|V_d|, |V_m|) = 2\left|\frac{|V_d| - |V_m|}{|V_d| + |V_m|}\right|$$

Optima employs the integrated similarity mentioned in Section 1.3 to evaluate the lexical similarity $P_s(x_a, y_\alpha)$.

**M Step** The maximization step chooses the match matrix, $M_*^{n+1}$, that maximizes $Q(M^{n+1}|M^n)$, as shown in Eq. 3. This mapping matrix becomes the input for the expectation step of the next iteration. Optima adopts the generalized EM, which relaxes maximization by settling for a mixture model, $M_*^{n+1}$, that simply improves the Q values.

$$M_*^{n+1} = M^{n+1} \in \mathcal{M} : Q(M^{n+1}|M^n) \geq Q(M^n|M^n) \tag{7}$$

The prior, $\pi_\alpha^{n+1}$, for each model graph node, $\alpha$, is updated as:

$$\pi_\alpha^{n+1} = \frac{1}{|V_d|} \sum_{\alpha=1}^{|V_d|} Pr\left(y_\alpha|x_a, M^n\right) \tag{8}$$

The updated $\pi_\alpha^{n+1}$ will be used in the next iteration of the EM.

### 1.3 Specific Techniques Used

We configured Optima slightly different for OAEI from its default configuration.

**Integrated Similarity Measure** Concept or word similarity measures may be broadly categorized into syntactic and semantic. Syntactic similarity between concepts is entirely based on the string similarity between the concepts' names, labels and other associated text. Semantic similarity measures attempt to utilize the meaning behind the concept names to ascertain the similarity of the concepts. Optima utilizes both syntactic and semantic similarities.

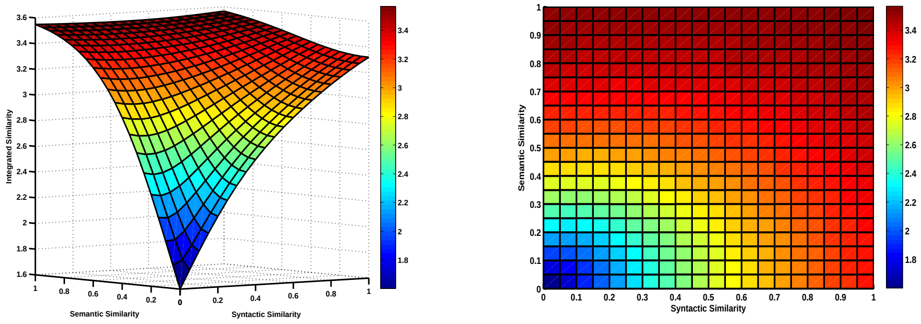

**Fig. 1.** Integrated similarity - 3D sigmoid

**Fig. 2.** Integrated similarity - Top angle

**Fig. 3.** Our integrated similarity measure as a function of the WN-based semantic similarity (*Sem*) and Smith-Waterman based syntactic similarity (*Syn*). Notice that the value is lower if semantic similarity is low but syntactic is high compared to vice versa.

There is no standard way of integrating WN-based similarity with syntactic measures. We employs a technique from our previous work in [8] to integrate similarity measures. We define a normalized 3D function that maps a given pair of semantic and syntactic similarity to an integrated value. In order to generate this function, we observe that labels that are syntactically similar (such as *cat* and *bat*) may have different meanings. Because we wish to meaningfully map entities, semantic similarity takes precedence over syntactic. Consequently, high syntactic but low semantic similarity results in a lower integrated similarity value in comparison to low syntactic but high semantic similarity. We model such an integrated similarity measure as shown in Fig. 3 and give the function in Eq. 9. Our integrated similarity function is similar to a 3D sigmoid restricted to the quadrant where the semantic and syntactic similarities range from 0 to 1. One difference from the exact sigmoid is due to the specific property it must have because semantic similarity takes precedence over syntactic. We used Lin [4] similarity measure and gloss-based cosine similarity measure to evaluate the semantic similarity. On the other hand we used Smith-Waterman [7] technique for ascertaining the syntactic similarity between concept and relationship names.

$$Int(x_a, y_\alpha) = \gamma \frac{1}{1 + e^{t \cdot r - c(Sem)}} \tag{9}$$

Here, $\gamma$ is a normalization constant; $r = \sqrt{Syn^2 + Sem^2}$, which produces the 3D sigmoid about the origin; $t$ is a scaling factor and $c(Sem)$ is a function of the semantic similarity as shown below: $c(Sem) = \frac{2}{1 + e^{t' \cdot Sem(x_a, y_\alpha) - c'}}$ where $t'$ is the scaling factor and $c'$ is the translation factor, if needed. The specific function in Fig. 3 is obtained when $t = 4$, $t' = 3.5$, and $c' = 2$.

### 1.4 Adaptations made for the evaluation

The iterative alignment algorithm requires a seed map. This is an initial list of mappings between concepts often provided to iterative algorithms. While the seed map could be generated manually, Optima additionally utilizes a simple technique of mapping nodes across the ontologies whose labels are syntactically similar. Candidate alignments are generated using simple but intuitive heuristics. For example, given each previously mapped node pair, their parents are considered for a match. Additionally, their sibling nodes could be considered. Analogous to the seed map, node pairs among the parents that are sufficiently similar are matched. Different potential alignments are generated based on how many parent nodes are matched and whether siblings are matched as well. These candidate alignments are considered during each iteration of Optima. More details about Optima are available in [2].

We also relaxed the Optima 's many-to-one constrain in candidate alignment generation to generate many-to-many alignments for OAEI.

### 1.5 Link to the system and parameters file

The Optima can be found at `http://thinc.cs.uga.edu/thinclabwiki/index.php/Automated_Alignment_of_Ontologies`.

### 1.6 Link to the set of provided alignments (in align format)

The OAEI 2011 results can be found at `http://thinc.cs.uga.edu/thinclabwiki/index.php/OAEI_2011`.

## 2 Results

As stated above, Optima participated in three tracks in OAEI 2011. However for this report preliminary results of two tracks are presented and the related analysis are reported.

### 2.1 Benchmark

The average precision and recall of Optima are depicted in 1.

| | **Precision** | **Recall** |
|---|---|---|
| 100 | 0.90 | 1.0 |
| 200 | 0.79 | 0.73 |
| 300 | 0.74 | 0.79 |

**Table 1.** Recall and Precision of Optima on benchmark track

### 2.2 Conferences

Optima attains an average recall of 0.60 and an average precision of 0.26 in conference track. See Appendix A for details.

### 2.3 Anatomy

We could not produce the results for anatomy track using Optima within the provided time. Since Optima utilizes an iterative algorithm and anatomy track has very large ontologies, we were unable to complete aligning these ontologies.

## 3 General comments

The primary challenge for Optima is to align very large ontologies. Due to its iterative nature and inherent computational complexity of evaluating the Equation 3, Optima takes considerably longer time to align larger ontologies. However it is able to align small to medium ontologies competitively.

We also found that computing semantic similarity measures for word phrases and compound words is difficult. Tokenizing these correctly and locating individual glosses in WN is often challenging[1] but crucial for a better performance.

[1] The concept *Meta-Review* should be tokenized into two words *(Meta, Review)* while *Registration_Non–Member* needs to be tokenized into two words *(Registration, NonMember)* but

## 4 Conclusion

In this report we present the results of Optima in OAEI 2011 campaign. We participate in three tracks including Benchmark, Conference and Anatomy. We reviewed the iterative algorithm Optima adopts to arrive at an inexact match between two ontologies. Though we have been using OAEI datasets for various experiments and fine tuning of Optima , this is the first time we participate officially in an OAEI campaign. Due to its iterative nature Optima takes substantially longer time to align large ontologies. As a result we are unable to provide our preliminary results of anatomy track for this report. In future, we would like to participate in more tracks. Especially we hope to leverage Optima to be able to efficiently solve instance matching and large ontology matching challenges.

## References

1. Dempster, A.P.; Laird N.M. and Rubin, D.B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)* 39:1 – 38.
2. Doshi, P.; Kolli, R.; and Thomas, C. 2009. Inexact matching of ontology graphs using expectation-maximization. *Web Semantics* 7(2):90 – 106.
3. Hayes, J. and Gutierrez, C. 2004. Bipartite graphs as intermediate models for RDF graphs. *Proceedings of the International Semantic Web Conference (ISWC)* :47 – 61.
4. Lin, D. 1998. An information-theoretic definition of similarity. In *ICML*, 296–304.
5. Luo, B. and Hancock, E. 2001. Structural graph matching using the EM algorithm and singular value decomposition. *Graph Algorithms and Computer Vision* 23 10:1120 – 1136.
6. McGuinness, D., and Harmelen, F. 2004. Owl web ontology language overview. Recommendation, World Wide Web Consortium (W3C).
7. Smith, T. F., and Waterman, M. S. 1981. Identification of common molecular subsequences. *Journal of molecular biology* 147(1):195–197.
8. Uthayasanker, T., and Doshi, P. 2011. On the Utility of WordNet for Ontology Alignment: Is it Really Worth It?. *IEEE ICSC* :268–274.

should not be tokenized into three words *(Registration, Non, Member)*. The hyphen (–) is a delimiter in the former concept but should be just ignored in the later concept. This tokenization is demanded by WN matchers since *MetaReview* does not exist in WN but the word *NonMember* exists in WN.

## A Optima 's performance in conference track

The precision and recall for individual test cases in conference track is shown tn the table 2 below.

| Ontology pair | Precision | Recall |
|---|---|---|
| cmt-confOf | 0.35 | 0.50 |
| cmt-conference | 0.18 | 0.44 |
| cmt-edas | 0.24 | 0.69 |
| cmt-ekaw | 0.15 | 0.45 |
| cmt-iasted | 0.33 | 1.00 |
| cmt-sigkdd | 0.39 | 0.75 |
| confOf-edas | 0.27 | 0.68 |
| confOf-ekaw | 0.30 | 0.55 |
| confOf-iasted | 0.33 | 0.67 |
| confOf-sigkdd | 0.26 | 0.71 |
| conference-confOf | 0.32 | 0.67 |
| conference-edas | 0.17 | 0.53 |
| conference-ekaw | 0.16 | 0.40 |
| conference-iasted | 0.15 | 0.29 |
| conference-sigkdd | 0.34 | 0.67 |
| edas-ekaw | 0.21 | 0.52 |
| edas-iasted | 0.35 | 0.47 |
| edas-sigkdd | 0.26 | 0.60 |
| ekaw-iasted | 0.20 | 0.60 |
| ekaw-sigkdd | 0.27 | 0.64 |
| iasted-sigkdd | 0.31 | 0.73 |
| average | 0.26 | 0.60 |

**Table 2.** Optima 's performance in conference track of OAEI 2011