Coherent Top-k Ontology Alignment for OWL EL

Jan Noessner, Mathias Niepert, and Heiner Stuckenschmidt

KR & KM Research Group,
Universität Mannheim
Mannheim, Germany
{jan,mathias,heiner}@informatik.uni-mannheim.de

Abstract. The integration of distributed information sources is a key challenge in data and knowledge management applications. Instances of this problem range from mapping schemas of heterogeneous databases to object reconciliation in linked open data repositories. In this paper, we approach the problem of aligning description logic ontologies. We focus particularly on the problem of computing coherent alignments, that is, alignments that do not lead to unsatisfiable classes in the resulting merged ontologies. We believe that considering coherence during the alignment process is important as it is this logical concept that distinguishes ontology alignment from other data integration problems. Depending on the heterogeneity of the ontologies it is often more reasonable to generate alignments with at most k correspondences because not every entity has a matchable counterpart. We describe both greedy and optimal algorithms for computing coherent top-k alignments between OWL EL ontologies and assess their performance relative to state-of-the-art matching systems.

1 Introduction

The growing number of heterogeneous knowledge bases on the web has made data integration systems a key technology for sharing and accumulating distributed information sources. In this paper, we focus on the problem of aligning description logic ontologies. Due to the explicit semantics of ontologies, alignment systems can take advantage of the logical concepts of coherence and consistency. Ensuring complete coherency and consistency is especially important in the area of ontology merging, where two ontologies are merged to one single ontology using the generated reference alignment.

Ontology debugging, for instance, is the process of efficiently finding and eliminating incoherencies. Several approaches to this problem were presented in [21] where the debugging process was based on the computation of minimal conflict sets. Similar concepts and algorithms have been used to debug precomputed ontology alignments [13]. Their algorithm scales well for few conflict sets in the alignment, but if the number of conflict sets increase, the performance decreases significantly. In [15] they build a set of hard and soft markov logic rules to reduce the incoherency of the alignment. Although the performance is still high for many conflict sets and most of the incoherencies are filtered out, the delivered alignments are not guaranteed to be coherent. In both, [13] and [15] a threshold is used to pre-select correspondences and a reasoner is needed to pre-calculate certain axioms.

Currently, most state-of-the-art matching systems such as Falcon [10], Aroma [5], and AgreementMaker [4] generate incoherent alignments [7]. To the best of our knowledge, only two of the matching systems that participated in the ontology alignment evaluation initiative (OAEI) of 2010 reduce the degree of alignment incoherence. While the semantic verification algorithm [11] of ASMOV reduces incoherence in a post-processing step CODI [17] employs incoherence reducing rules during the alignment process. Both matching systems, however, do not guarantee the final alignments to be coherent [7]. Another matching system not participating in the OAEI but focusing on coherent alignments is PROMPT [18]. It provides the user with different interactive views on the ontologies and aids the merging process by pointing out logical conflicts.

Depending on the heterogeneity of the ontologies it is often more sensible to generate alignments with at most k correspondences because not every entity in one ontology has a matchable counterpart in the other. Top-k algorithms are common in the area of information retrieval and ranking and have recently been applied in more structured data management systems. In the context of database schema matching, for instance, [8] presented an approach to computing the best k schema mappings.

With this paper, however, we present an *optimal coherent* top-k ontology matching algorithm, that is, an algorithm that generates optimal coherent alignments of size at most k. Compared to [13] our approach will still perform well for large number of conflict sets. The strength of the approach lies in its ability to incorporate arbitrary confidence values which could have been for example computed by other matching applications. Hence, the top-k algorithms are not intended to compete with existing matching systems but rather to complement their strength in deriving high-quality confidence values.

We present both a greedy and an optimal algorithm for computing coherent top-k alignments. The optimal algorithm utilizes the existence of a set of materialization rules for the description logic $\mathcal{EL}^{++}$ [1, 12] without nominals and concrete domains, and formulates the alignment tasks as linear optimization problems. To reduce the complexity of these problems, the algorithm combines a cutting plane inference and a delayed column generation algorithm originally developed in the context of Markov logic [19].

We conduct extensive experiments to evaluate the accuracy and efficiency of both the greedy and optimal top-k algorithms. We also compare the coherence, recall, precision, and F_1 scores of the computed alignments with those generated by various state-of-the-art matching systems.

Name	Syntax	Semantics
top	$\top$	$\Delta^{\mathcal{I}}$
bottom	$\bot$	$\emptyset$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
GCI	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
RI	$r_1 \circ ... \circ r_k \sqsubseteq r$	$r_1^{\mathcal{I}} \circ ... \circ r_k^{\mathcal{I}} \subseteq r^{\mathcal{I}}$

Table 1. The description logic $\mathcal{EL}^{++}$ without nominals and concrete domains.

2 Description Logics

Description logics (DLs) are a family of knowledge representation languages [3]. They provide the logical formalism for ontologies and the Semantic Web. We focus on the DL $\mathcal{EL}^{++}$ which captures the expressivity of numerous real-world ontologies. $\mathcal{EL}^{++}$ is the description logic on which the web ontology language profile OWL 2 EL is based [1]. Reasoning tasks such as consistency and instance checking can be performed in polynomial time. Therefore, $\mathcal{EL}^{++}$ is practical for applications employing ontologies with large numbers of properties and classes. It is possible to express disjointness of complex concept descriptions as well as range and domain restrictions [2] and role inclusion axioms (RIs) allow the expression of role hierarchies $r \sqsubseteq s$ and transitive roles $r \circ r \sqsubseteq r$.

$\mathcal{EL}^{++}$ concept descriptions are defined recursively by a set of constructors, starting with a set $\mathsf{N_C}$ of concept names, a set $\mathsf{N_R}$ of role names, and a set $\mathsf{N_I}$ of individual names. Concept descriptions and role inclusions in $\mathcal{EL}^{++}$ are built with the constructors depicted in Table 1. We write r, s to denote role names and C, D to denote concept descriptions. The semantics of the concept descriptions in $\mathcal{EL}^{++}$ are defined in terms of an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$. The interpretation function $\cdot^{\mathcal{I}}$ is recursively defined as shown in Table 1. A concept C *is subsumed by* a concept D with respect to a CBox $\mathcal{C}$, written $C \sqsubseteq_{\mathcal{C}} D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ in every model $\mathcal{I}$ of $\mathcal{C}$.

A constraint box (CBox) is a finite set of general concept inclusions (GCIs) and role inclusions (RIs). Given a CBox $\mathcal{C}$, we use $\mathsf{BC}_{\mathcal{C}}$ to denote the set of *basic concept descriptions*, that is, the smallest set of concept descriptions consisting of the top concept $\top$, all concept names used in $\mathcal{C}$, and all nominals $\{a\}$ appearing in $\mathcal{C}$. Then, $\mathcal{C}$ is in normal form if all GCIs have one of the following forms, where $C_1, C_2 \in \mathsf{BC}_{\mathcal{C}}$ and $D \in \mathsf{BC}_{\mathcal{C}} \cup \{\bot\}$:

$$\begin{array}{ll} C_1 \sqsubseteq D; & C_1 \sqsubseteq \exists r.C_2; \\ C_1 \sqcap C_2 \sqsubseteq D; & \exists r.C_1 \sqsubseteq D \end{array}$$

and if all role inclusions are of the form $r \sqsubseteq s$ or $r_1 \circ r_2 \sqsubseteq s$. By applying a finite set of rules and introducing new concept and role names, any CBox

$\mathcal{C}$ can be turned into a normalized CBox [1]. For any $\mathcal{EL}^{++}$ CBox $\mathcal{C}$ we write $\mathsf{norm}(\mathcal{C})$ to denote the set of normalized axioms that result from the application of the normalization rules to $\mathcal{C}$. A normalized $\mathcal{EL}^{++}$ CBox is *classified* when subsumption relationships between *all* concept names are made explicit. A CBox $\mathcal{C}$ is *coherent* if for all concept names C in $\mathcal{C}$ we have that $C \not\sqsubseteq_{\mathcal{C}} \bot$.

3 Coherent Ontology Alignment

Ontology alignment is the process of inferring correspondences between entities of two ontologies. We begin by formally defining the notions of *correspondence* and *alignment* based on a definition by Euzenat and Shvaiko [6]. In this paper, each *ontology* is equivalent to a $\mathcal{EL}^{++}$ CBox *without* nominals and concrete domains, that is, an OWL 2 EL ontology without nominals and datatype properties. We refer the reader to [9] for a primer of the W3C recommendation for OWL 2 and its profiles.

Definition 1 (Correspondence and Alignment). *Given ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, let q be a function that defines sets of matchable entities $q(\mathcal{O}_1)$ and $q(\mathcal{O}_2)$. A correspondence between $\mathcal{O}_1$ and $\mathcal{O}_2$ is a triple $\langle e_1, e_2, r\rangle$ such that $e_1 \in q(\mathcal{O}_1)$, $e_2 \in q(\mathcal{O}_2)$, and r is a semantic relation. An alignment between $\mathcal{O}_1$ and $\mathcal{O}_2$ is a set of correspondences between $\mathcal{O}_1$ and $\mathcal{O}_2$.*

The general form of Definition 1 captures a wide range of correspondence types. In the following we focus on equivalence correspondences between concepts and object properties, respectively. The majority of matching systems provide normalized confidence values for each correspondence. Based on these confidence values, an alignment is extracted by applying a threshold $\tau \in [0, 1]$ meaning that only the correspondences with a confidence value greater than or equal to τ are included in the alignment.

In this paper, however, we are interested in solutions to the problem of computing *coherent alignments* between ontologies. An alignment $\mathcal{A}$ is coherent with respect to the coherent ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ if the ontology $\mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{A}$ is coherent, that is, if the ontology that results from merging $\mathcal{O}_1$ and $\mathcal{O}_2$ under the alignment $\mathcal{A}$ is coherent. Hence, in the remainder of the paper, we assume the existence of confidence values provided by, for instance, state-of-the-art matching systems. We refer to these values as *a-priori* confidence values. The *score* of an alignment is the sum of confidence values of its correspondences. We say that an alignment $\mathcal{A}$ of size k with score s is *optimal* if for every other alignment of size at most k with score s' we have that $s' \leq s$.

3.1 Greedy Coherent Top-k Alignment

The first algorithm for generating top-k coherent alignments from a set of correspondences with a-priori confidence values follows a greedy strategy. It appends an initially empty alignment with correspondences according to their a-priori confidence values in descending order. After each addition, it employs a reasoner

F_1	$\forall c : sub(c, c)$
F_2	$\forall c : sub(c, \top)$
F_3	$\forall c, c', d : sub(c, c') \wedge sub(c', d) \Rightarrow sub(c, d)$
F_4	$\forall c, c_1, c_2, d : sub(c, c_1) \wedge sub(c, c_2) \wedge int(c_1, c_2, d) \Rightarrow sub(c, d)$
F_5	$\forall c, c', r, d : sub(c, c') \wedge rsup(c', r, d) \Rightarrow rsup(c, r, d)$
F_6	$\forall c, r, d, d', e : rsup(c, r, d) \wedge sub(d, d') \wedge rsub(d', r, e) \Rightarrow sub(c, e)$
F_7	$\forall c, r, d, s : rsup(c, r, d) \wedge psub(r, s) \Rightarrow rsup(c, s, d)$
F_8	$\forall c, r_1, r_2, r_3, d, e : rsup(c, r_1, d) \wedge rsup(d, r_2, e) \wedge pcom(r_1, r_2, r_3) \Rightarrow rsup(c, r_3, e)$
F_9	$\forall c : \neg sub(c, \bot)$
F_{10}	$\forall c_1, c_2 : cmap(c_1, c_2) \Rightarrow sub(c_1, c_2)$
F_{11}	$\forall c_1, c_2 : cmap(c_1, c_2) \Rightarrow sub(c_2, c_1)$
F_{12}	$\forall r_1, r_2 : pmap(r_1, r_2) \Rightarrow psub(r_1, r_2)$
F_{13}	$\forall r_1, r_2 : pmap(r_1, r_2) \Rightarrow psub(r_2, r_1)$

Table 2. The first-order theory $\mathcal{F}$. Valid instantiations of the formulas are those compatible with the types of the predicates from Definition 2. The predicates *cmap* and *pmap* model the correspondences between concept and role names, respectively. $\bot$ and $\top$ are constant symbols representing the bottom and top concept.

to check whether the resulting alignment causes incoherences, and if it does, removes the previously added correspondence. The advantage of the approach is its efficiency – classification and, therefore, checking coherence of OWL EL ontologies can be performed in polynomial time. However, the approach does not compute optimal alignments. Once a correspondence has been added it cannot be revoked in later stages of the computation. The following example from the conference domain demonstrates said problem.

Example 1. Let $\mathcal{O}_1$ contain the axiom $Review \sqcap JournalReviewer \sqsubseteq \bot$ and $\mathcal{O}_2$ the axiom $Reviewer \sqcap PaperReview \sqsubseteq \bot$. Moreover, consider the following correspondences and their associated a-priori confidence values: $\langle Reviewer \equiv Review, 0.9\rangle$, $\langle PaperReview \equiv Review, 0.7\rangle$, $\langle Reviewer \equiv JournalReviewer, 0.6\rangle$. The greedy top-$k$ approach would include the correspondence $\langle Reviewer \equiv Review, 0.9\rangle$ and would not add more correspondences due to the resulting incoherence. While an optimal top-k approach would also add the same correspondence for $k = 1$ it would generate the correct alignment $\{\langle PaperReview \equiv Review, 0.7\rangle, \langle Reviewer \equiv JournalReviewer, 0.6\rangle\}$ for $k = 2$ revoking the previous decision.

In the following we introduce a novel algorithm that computes optimal coherent top-k alignments. It leverages the completion rules for the DL $\mathcal{EL}^{++}$ without nominals and concrete domains [1, 12].

3.2 Optimal Coherent Top-k Alignment

The *optimal* top-k alignment algorithm which we describe in the remainder of this section computes an *optimal coherent* alignment of size at most k from a given set of a-priori confidence values. The crucial insight is that the optimal alignment problem can be reduced to an optimization problem: Given the a-priori confidence values and the two input ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, *maximize* the sum of confidence values of correspondences in the alignment *subject to* the coherence of the ontology that results when merging $\mathcal{O}_1$ and $\mathcal{O}_2$ under the alignment. In order to guarantee the coherence of the alignment we map the normalized axioms of the two ontologies to ground predicates and formulate the optimization problem in such a way that all solutions to the problem correspond to coherent ontologies. We achieve this through a set of materialization formulas that capture the underlying DL semantics. We refer the reader to [12] for more details on materialization calculi and to [1, ?] for the completeness of a finite set of completion rules for $\mathcal{EL}^{++}$ from which the set of formulas $\mathcal{F}$ (see Table 2) is partially derived. Furthermore, we refer the reader to [16] for the introduction of log-linear description logic which is the foundation of our approach. We begin by defining the mapping φ between ontologies and sets of ground atoms of the theory $\mathcal{F}$.

Definition 2 (Ontology Transformation). *Let $\mathcal{O}_1$ and $\mathcal{O}_2$ be two normalized ontologies, let $\mathsf{N_U} = \mathsf{BC}_{\mathcal{O}_1} \cup \mathsf{BC}_{\mathcal{O}_2}$ be the set of basic concept descriptions of both ontologies, and $\mathcal{H}$ be the set of all valid instantiations of predicates in $\mathcal{F}$ (see Table 2) relative to $\mathsf{N_U}$ (the Herbrand base of $\mathcal{F}$ with respect to $\mathsf{N_U}$ as a set of constant symbols). The function φ maps $\mathcal{O}_1 \cup \mathcal{O}_2$ to a subset of $\mathcal{H}$ as follows.*

$$\begin{array}{ll}
C_1 \sqsubseteq D & \mapsto sub(c_1, d) \\
C_1 \sqcap C_2 \sqsubseteq D & \mapsto int(c_1, c_2, d) \\
C_1 \sqsubseteq \exists r.C_2 & \mapsto rsup(c_1, r, c_2) \\
\exists r.C_1 \sqsubseteq D & \mapsto rsub(c_1, r, d) \\
r \sqsubseteq s & \mapsto psub(r, s) \\
r_1 \circ r_2 \sqsubseteq r_3 & \mapsto pcom(r_1, r_2, r_3).
\end{array}$$

All predicates are typed meaning that $r, s, r_i, (1 \leq i \leq 3)$, are role names, C_1, C_2 basic concept descriptions, and D basic concept descriptions or the bottom concept.

Based on the previously defined mapping, we can state the computation of an optimal top-k alignment as an instance of integer linear programming (ILP). Let $\mathcal{O}_1$ and $\mathcal{O}_2$ be two normalized *coherent* ontologies, let $\mathsf{N_U} = \mathsf{BC}_{\mathcal{O}_1} \cup \mathsf{BC}_{\mathcal{O}_2}$, let $\mathcal{F}^{\mathsf{N_U}}$ be the set of all valid instantiations of $\mathcal{F}$ relative to $\mathsf{N_U}$, and let $\mathcal{H}$ be the Herbrand base of $\mathcal{F}$ relative to $\mathsf{N_U}$. Moreover, let $\mathcal{K} = \varphi(\mathcal{O}_1 \cup \mathcal{O}_2)$ and let $\mathcal{L}$ be a set of valid instantiations of the predicates *cmap* and *pmap* modeling correspondences between classes and object properties, respectively, each associated with its a-priori confidence.

For each ground atom g_i occurring at least once in either $\mathcal{L}$ (with a-priori confidence value w_i), $\mathcal{K}$, or in a formula in $\mathcal{F}^{\mathsf{N_U}}$ we associate a variable $x_i \in \{0, 1\}$.

Let C^L be the set of indices of ground atoms in $\mathcal{L}$, let C^K be the set of indices of ground atoms in $\mathcal{K}$, and let C_j^F ($\bar{C}_j^F$) be the set of indices of unnegated (negated) ground atoms in the clause equivalent to $F_j \in \mathcal{F}^{\mathsf{N}_\cup}$. Then, the *top-k ILP with respect to* $\mathcal{L}$ is stated as follows

$$\textbf{max} \sum_{i \in C^L} w_i x_i \qquad \textbf{subject to} \qquad \sum_{i \in C^L} x_i \leq k \;\; \textbf{and}$$

$$\sum_{i \in C^K} x_i \geq |C^K| \;\; \textbf{and} \sum_{i \in C_j^F} x_i + \sum_{i \in \bar{C}_j^F} (1 - x_i) \geq 1, \; \forall j \qquad (2).$$

Theorem 1. *Each solution of the Top-k ILP with respect to $\mathcal{L}$ corresponds to an ontology that results from (a) merging the ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ under an optimal alignment $\mathcal{A} \subseteq \mathcal{L}$ of size at most k and (b) classifying the merged ontology.*

Thus, the algorithm not only computes an optimal coherent top-k alignment but also classifies the merged ontologies making subsumption relationships between each pair of classes explicit. For a proof concerning the classification and the coherency of Theorem 1 the reader is referred to [16].

The immediate addition of all above constraints, however, would result in a very complex and potentially intractable optimization problem. In order to avoid this problem, we combine variants of the cutting plane inference algorithm [20] and the delayed column generation algorithm [14] both of which were first proposed for computing maximum a-posteriori (MAP) states in Markov logic networks [19]. To compute the solution of a top-k ILP we first construct the top-k ILP *with respect to* the set $\mathcal{L}'$ containing only $m \geq k$ correspondences with highest a-priori confidence values. The ILP is initially solved *without* the constraints of type (2). Given the current solution, the algorithm determines all violated constraints of type (2) in polynomial time, adds those to the ILP, and solves the updated problem. This is repeated until no violated constraints remain. If the solution contains k correspondences we have found an optimal top-k alignment. Otherwise, the set $\mathcal{L}'$ is augmented with m more correspondences with highest a-priori confidence values and the top-k ILP *with respect to* $\mathcal{L}'$ is solved as before. This is repeated until we have found a solution with k correspondences or until $\mathcal{L}'$ contains *all* correspondences.

Due to the extendability of the ILP formulation of the top-k alignment problem it is possible to include additional types of constraints such as constraints enforcing functional and one-to-one alignments and constraints modeling known correct correspondences.

4 Experimental Evaluation

We conducted extensive experiments to evaluate the performance of the greedy and optimal top-k alignment algorithms. In particular, we compared the optimal with the greedy top-k algorithm both in terms of computation time and alignment accuracy. We also assessed the accuracy of the alignments by comparing

Axiom type	C	P	$C_1 \sqsubseteq D$	$C_1 \sqcap C_2 \sqsubseteq D$	$C_1 \sqsubseteq \exists r.C_2$	$\exists r.C_1 \sqsubseteq D$	$r \sqsubseteq s$	$r_1 \circ r_2 \sqsubseteq s$
Conference ontologies								
cmt	30	49	25	27	0	48	0	0
conference	60	46	56	14	7	47	13	0
confof	39	13	42	43	9	11	0	1
edas	104	30	90	409	3	29	0	0
ekaw	73	33	80	74	6	20	8	3
iasted	141	38	291	3	126	49	0	0
sigkdd	50	17	59	0	15	23	0	0
Anatomy ontologies								
mouse_anatomy	2744	3	4493	0	1637	0	0	0
nci_anatomy	3304	2	5423	17	1662	0	0	1

Table 3. Number of classes and properties as well as number of normalized EL axioms in the respective ontologies we used for the experiments.

them to the alignments generated by state-of-the-art matching systems that participated in the latest OAEI of 2010 [7]. Moreover, we analyzed and compared the degree of coherence of each of the alignments computed by the matching systems.

4.1 Experimental Set-Up

For the experimental evaluation we used the ontologies of the conference and anatomy tracks of the OAEI. The availability of reference alignments and recent results from state-of-the-art matching systems make the two tracks particularly suitable. The conference track consists of several expressive ontologies modeling the domain of scientific conferences. The ontologies have been developed by different groups and, therefore, reflect different conceptualizations of the same domain. Reference alignments for seven of these ontologies are made available by the organizers of the OAEI. These 21 alignments contain correspondences between concepts and properties including a reasonable number of non-trivial instances. The two ontologies of the anatomy track are from the medical domain modeling the anatomy of humans and mice, respectively, and consist of over 2500 classes each. Since our matching approach is restricted to EL axioms we used the OWL API to downgrade the more expressive conference ontologies. We applied the set of rules from [1, 2] to normalize the ontologies and to also include existing range restrictions. Table 3 lists the resulting conference and the anatomy ontologies along with the number of classes, properties, and normalized EL axioms.

We have argued that the top-k algorithms are not intended to compete with existing matching systems but rather to complement their strengths in gener-

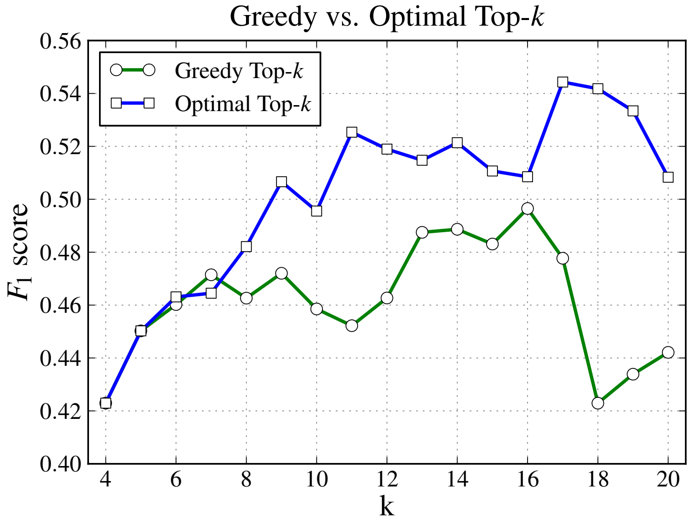

Fig. 1. F_1 scores of the optimal top-k and the greedy top-k algorithms averaged over the 21 alignment problems in the conference ontologies.

ating high-quality a-priori confidence values. Hence, in order to asses the algorithms' ability to compute alignments from given confidence values we used the Levenshtein distance normalized to the range $[-1, 1]$. Using such a naïve algorithm to derive confidence values lets us evaluate the performance of the alignment algorithm without being influenced by highly sophisticated confidence measures. Please note, however, that the strength of the approach is the ability to incorporate confidence values generated by existing matching systems. We employed the reasoner Pellet [22] for the greedy algorithm and the mixed ILP solver Gurobi[1] for the optimal algorithm. We also augmented the ILP with constraints enforcing functional one-to-one alignments and we set the parameter m to $2k$. The experiments were run on a desktop PC with AMD Athlon Dual Core Processor 5400B with 2.6GHz and 1GB RAM. The source files and supplementary materials are available at `http://code.google.com/p/elmatch/`.

4.2 Results of the Evaluation

We first assessed the relative performance of the two top-k algorithms with respect to their F_1 score. Figure 1 shows the F_1 scores of the optimal and greedy top-k algorithms averaged over the 21 ontology pairs of the conference track. For $k \leq 6$ the F_1 scores are almost identical which is due to the absence of incoherence causing correspondences in the small alignments. With $k = 8$, however, the optimal algorithm starts to outperform the greedy approach as the larger alignments cause incoherences and substitutions of correspondences of the type described in Example 1 are becoming more prevalent.

[1] http://www.gurobi.com/

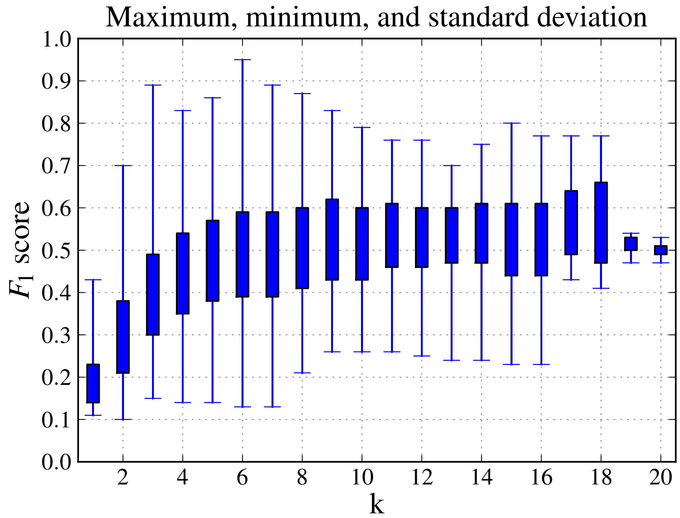

Fig. 2. Minimum, maximum and standard deviation of the F_1 score for the optimal top-k algorithm on the conference ontologies. The decrease in standard deviation for $k \geq 19$ is due to the fact that there are only few pairs of ontologies with functional one-to-one alignments of size k.

The runtime of the algorithms is summarized in Table 4. For the conference ontologies and $k \leq 10$ the run time of the optimal algorithm is comparable to the greedy approach. The reason for the increase in runtime of the optimal algorithm for $k = 20$ is caused by the small size of the ontologies – alignments of size 20 exist only between 9 of the 21 pairs of ontologies. Hence, the optimal algorithm has to include *all* correspondences in its ILP formulation thus increasing the complexity of the optimization problem. Interestingly, the effect is reversed for the anatomy ontologies. While the optimal algorithm has an overhead of about 40 seconds for classifying the large merged ontology the increase in runtime is smaller compared to the greedy approach. For $k = 1$ the greedy approach is about 10 times faster but only about twice as fast for $k = 20$. Considering that the reasoner Pellet is highly optimized for EL ontologies we find this to be a convincing result.

A suitable choice for the parameter k of the top-k algorithms clearly depends on the number of matchable elements and, therefore, on the size of the involved ontologies. Figure 2 shows the minimum, maximum and standard deviation of the optimal algorithm for the 21 different alignments. The large standard deviation and the discrepancy between the minima and maxima makes it evident that we need to adjust the parameter k *individually* for each alignment instance. We used the following *ad-hoc* heuristic to determine a suitable choice for the parameter k. We first computed the number P of correspondences where both matchable elements have identical labels. We then computed the parameter k with the formula $k = P + \alpha(k_{max} - P)$ where k_{max} is the maximal possible number of correspondences and $\alpha \in [0, 1]$. The parameter α determines the fraction of "non-trivial" correspondences one wants to derive and depends on the het-

k	1	5	10	15	20
Conference ontologies					
Greedy Top-k	0.36	0.41	0.56	0.77	1.21
Optimal Top-k	0.49	0.49	1.21	2.93	14.24
Anatomy ontologies					
Greedy Top-k	4.67	4.96	10.39	17.66	29.68
Optimal Top-k	40.76	42.42	45.10	48.74	53.60

Table 4. The average time in seconds needed to compute coherent top-k alignments for the benchmark and anatomy ontologies *and* classifying the merged ontologies.

erogeneity of the involved ontologies. In our experiments, we set the parameter α to 0.2. Table 5 depicts precision, recall, and F_1 scores of the optimal coherent top-k algorithm and a selection of matching systems that participated in the OAEI[2] with standard threshold 0.5. The coherent top-k algorithm has the best precision and competitive F_1 scores.

The main advantage of both top-k approaches compared to other matching systems, however, is the coherence of their alignments for EL ontologies. Table 5 lists the fraction of coherent alignments and classes, respectively, in the merged ontologies. Except for ASMOV, whose incomplete semantic verification algorithm [11] also reduces incoherences, all other matching systems generated incoherent alignments. In summary, only 14% of Aroma's, 29% of Falcon's, and 38% of AgreementMakers alignments were coherent indicating that these systems do not leverage the notion of coherence during the alignment process.

[2] Please visit `http://oaei.ontologymatching.org/2010/` for a complete list of results and all matching systems that participated at the OAEI 2010.

Matcher	Top-k	Falcon	AgrMaker	Aroma	ASMOV
Precision	0.78	0.59	0.50	0.36	0.45
Recall	0.44	0.58	0.65	0.49	0.07
F_1 score	0.57	0.58	0.57	0.42	0.12
Coh. Align	1.0	0.29	0.38	0.14	1.0
Coh. Class	1.0	0.95	0.84	0.64	1.0

Table 5. Comparison of the optimal top-k algorithm with state-of-the-art matching systems on the conference ontologies. Precision, recall, and F_1 scores are measured relative to the reference alignments. *Coh. Align* is the fraction of coherent alignments and *Coh. Class* is the fraction of coherent classes relative to the number of classes in all ontologies.

5 Conclusion & Future Work

With this paper, we presented a greedy and a novel optimal algorithm for computing coherent top-k alignments between OWL EL ontologies. The optimal algorithm employs integer linear programming solvers to maximize the sum of confidence values subject to the coherence of the ontology. Our evaluation showed that although we spent no effort on optimizing the confidence values (we used the simple Levenshtein distance), our F_1 scores were competitive compared to the participating systems at the OAEI 2010. The real strength of the top-k algorithms, however, is their ability to existing incorporate a-priori confidence values.

Currently, our approach is limited to the description logic $\mathcal{EL}^{++}$ without nominals and concrete domains but we intend to extend it to more expressive description logic languages such as Horn-$\mathcal{SHIQ}$. Moreover, we will work on supporting class and role assertions, nominals, and concrete domains. Apart from this, we will modify our approach to incorporate confidence values for complex correspondences. To this end, we will express complex matching patterns and their confidence values and integrate them in the optimization problem to compute coherent complex alignments between ontologies.

References

1. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Proceedings of the 19th International Joint Conference on Artificial Intelligence (2005)
2. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope further. In: Proceedings of the OWLED Workshop (2008)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook. Cambridge University Press (2003)
4. Cruz, I., Stroe, C., Caci, M., Caimi, F., Palmonari, M., Antonelli, F., Keles, U.: Using AgreementMaker to Align Ontologies for OAEI 2010. Proceedings of the 5th Workshop on Ontology Matching (2010)
5. David, J., Guillet, F., Briand, H.: Matching directories and OWL ontologies with AROMA. In: Proceedings of the 15th Conference on Information and knowledge management (2006)
6. Euzenat, J., Shvaiko, P.: Ontology matching. Springer-Verlag (2007)
7. Euzenat, J., et al.: First Results of the Ontology Alignment Evaluation Initiative 2010. Proceedings of the 5th Workshop on Ontology Matching (2010)
8. Gal, A.: Managing uncertainty in schema matching with top-k schema mappings. J. Data Semantics VI (2006)
9. Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P.F., Rudolph, S. (eds.): OWL 2 Web Ontology Language: Primer. W3C Recommendation (2009)
10. Hu, W., Chen, J., Cheng, G., Qu, Y.: ObjectCoref & Falcon-AO: Results for OAEI 2010. Proceedings of the 5th International Ontology Matching Workshop (2010)
11. Jean-Maya, Y.R., Shironoshitaa, E.P., Kabuka, M.R.: Ontology matching with semantic verification. Web Semantics 7(3) (2009)
12. Krötzsch, M.: Efficient inferencing for OWL EL. In: Proceedings of JELIA (2010)
13. Meilicke, C., Tamilin, A., Stuckenschmidt, H.: Repairing ontology mappings. In: Proceedings of the Conference on Artificial Intelligence (2007)

14. Niepert, M.: A Delayed Column Generation Strategy for Exact k-Bounded MAP Inference in Markov Logic Networks. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (2010)
15. Niepert, M., Meilicke, C., Stuckenschmidt, H.: A Probabilistic-Logical Framework for Ontology Matching. In: Proceedings of the 24th AAAI Conference on Artificial Intelligence (2010)
16. Niepert, M., Noessner, J., Stuckenschmidt, H.: Log-Linear Description Logics. In: Proceedings of IJCAI (2011)
17. Noessner, J., Niepert, M.: CODI: Combinatorial Optimization for Data Integration–Results for OAEI 2010. Proceedings of the 5th Workshop on Ontology Matching (2010)
18. Noy, N., Musen, M.: The PROMPT suite: interactive tools for ontology merging and mapping. International Journal of Human-Computer Studies 59(6), 983–1024 (2003)
19. Richardson, M., Domingos, P.: Markov logic networks. Machine Learning 62(1-2) (2006)
20. Riedel, S.: Improving the accuracy and efficiency of map inference for markov logic. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence (2008)
21. Schlobach, S., Huang, Z., Cornet, R., van Harmelen, F.: Debugging incoherent terminologies. J. Autom. Reasoning 39(3) (2007)
22. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: a practical OWL-DL reasoner. Journal of Web Semantics 5(2), 51–53 (2007)