Aligner automatiquement des ontologies avec oMAP

Raphaël Troncy*

*CWI Amsterdam, P.O. Box 94079, 1090 GB, The Netherlands
Raphael.Troncy@cwi.nl,
http://www.cwi.nl/~troncy/

1 Introduction

Nous nous intéressons aux ontologies décrites dans un même langage de représentation des connaissances (OWL) et nous proposons un cadre général nommé oMAP pour automatiquement aligner des ontologies OWL sur le web. **oMAP** (Straccia et Troncy, 2005a, 2006) permet de trouver les meilleures correspondances (avec leurs poids) entre des entités définies dans des ontologies, en combinant la prédiction de plusieurs classifieurs. Ceux-ci peuvent être terminologiques ou basés sur des techniques statistiques d'apprentissage, ou encore utilisent la sémantique des axiomes OWL pour établir des relations d'équivalence entre des classes et des propriétés définies dans les ontologies. oMAP est disponible à :
http://www.cwi.nl/~troncy/oMAP/.

Notre approche s'inspire des travaux formels menés autour de l'échange d'information et emprunte à d'autres approches comme GLUE (Doan et al., 2003) l'idée de combiner plusieurs composants spécialisés pour obtenir le meilleur résultat.

Théoriquement, un *mapping* est un tuple $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma)$ où $\mathbf{S}$ et $\mathbf{T}$ sont respectivement les ontologies source et cible, et Σ est un ensemble de règles d'appariement de la forme : $\alpha_{j,i}\, T_j \leftarrow S_i$ où S_i et T_j représentent respectivement une entité (classe ou propriété) de l'ontologie source et cible. Cette règle signifie que l'entité S_i de l'ontologie source correspond à l'entité T_j dans l'ontologie cible, et que la mesure de confiance associée à cet appariement est $\alpha_{j,i}$. Notons qu'aucune contrainte supplémentaire n'est posée sur ces mises en correspondance : un concept de l'ontologie source peut correspondre à 0 ou plusieurs concepts dans l'ontologie cible et réciproquement.

Aligner deux ontologies dans *oMAP* est un processus en trois étapes :

1. Nous devinons un ensemble Σ possible, et nous estimons sa qualité en fonction de la mesure de confiance associée à chacune des règles d'appariement qu'il contient.
2. Pour chaque règle $T_j \leftarrow S_i$, nous estimons le poids $\alpha_{i,j}$, qui dépend aussi de l'ensembleΣ, i.e. $\alpha_{i,j} = w(S_i, T_j, \Sigma)$.
3. Comme nous ne pouvons pas considérer tous les ensembles Σ possibles (il y en a un nombre exponentiel) pour ensuite choisir le meilleur, nous construisons itérativement cet ensemble et nous utilisons diverses heuristiques pour réduire la complexité.

De la même manière que GLUE (Doan et al., 2003), nous estimons le poids $w(S_i, T_j, \Sigma)$ d'une règle $T_j \leftarrow S_i$ en combinant les prédictions de différents classifieurs $CL_1, \ldots, CL_n$.

Chaque classifieur calcule un poids $w(S_i, T_j, CL_k)$, approximation de la règle $T_j \leftarrow S_i$ pour le classifieur CL_k. Une liste de priorité permet finalement de combiner toutes ces prédictions qui ne sont donc pas simplement moyennées (Figure 1).

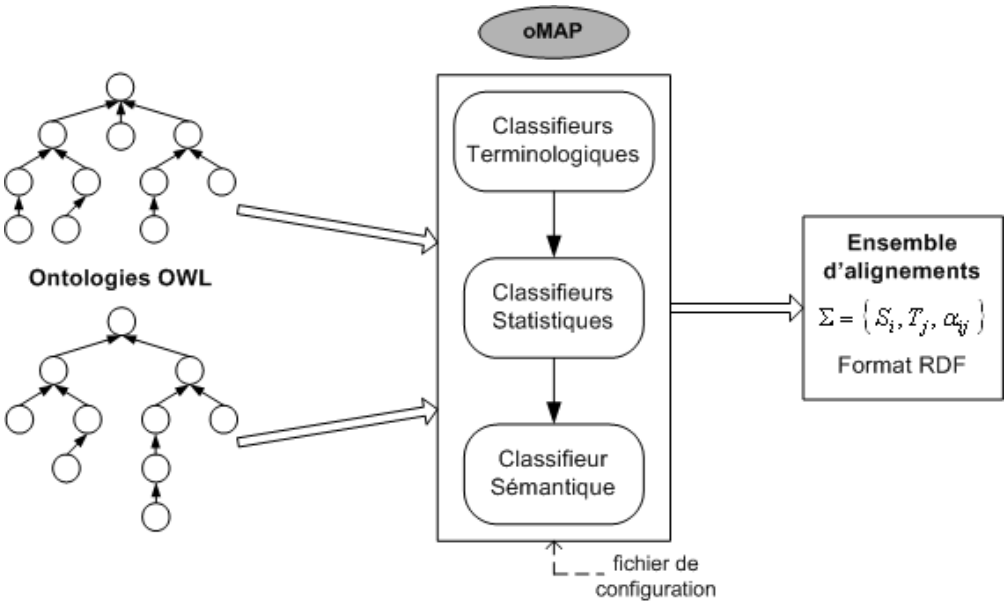

FIG. 1 – *Architecture générale de* oMAP. *Les différents classifieurs peuvent être combinés via un fichier de configuration.* oMAP *produit le meilleur ensemble d'alignements possibles, établissant une relation d'équivalence entre une entité i d'une ontologie S et une entité j d'une ontologie T avec la mesure de confiance α_{ij}.*

Nous décrivons dans la suite les différents classifieurs implémentés dans oMAP. Certains ne considèrent que la partie terminologique des ontologies alors que d'autres utilisent des méthodes statistiques d'apprentissage et peuvent utiliser les instances de l'ontologie. Finalement, un troisième type de classifieur fait appel à la structure et à la sémantique des définitions et des axiomes OWL.

2 Classifieurs terminologiques, statistiques et sémantique

Les classifieurs terminologiques utilisent le nom et les commentaires de chaque ressource ontologique. En OWL, chaque ressource est identifiée par un URI, mais peut également avoir des propriétés d'annotation attachées. Parmi celles-ci, la propriété `rdfs :label` est généralement utilisée pour fournir le nom usuel de la ressource. De plus, le multilinguisme des noms est supporté par l'attribut langue des littéraux RDF. Dans la suite, nous considérons que le nom d'une entité est donné par la valeur de la propriété `rdfs :label` ou par le fragment

de l'URI l'identifiant si cette propriété n'est pas spécifiée. Les classifieurs terminologiques de oMAP comparent le nom des entités, leur racine[1], ou calculent des mesures de similarité entre les chaînes de caractères (e.g. la distance de Levenshtein). Finalement, des dictionnaires ou ressources terminologiques tels que WordNet[2] peuvent être utilisés dans le calcul de la distance.

Les classifieurs statistiques considèrent les instances (ou individus) des ontologies. Les règles suivantes sont utilisées pour construire le texte correspondant à ces instances : nous considérons (i) le nom des individus nommés, (ii) la valeur pour les propriétés de type de données (`owl :DatatypeProperty`) et (iii) le type pour les individus anonymes et pour le co-domaine des propriétés de type objets (`owl :ObjectProperty`). Par exemple,, considérons les individus suivants :

Individual (x_1 type `(Workshop)`
  value `(label "Workshop DECOR")`
  value `(location` x_2`))`
Individual (x_2 type `(Address)`
  value `(city "Namur")` value `(country "Belgique"))`

Le texte u_1 pour l'individu x_1 sera donc (`"Workshop DECOR", "Address"`) et le texte u_2 pour l'individu anonyme x_2 sera (`"Address", "Namur", "Belgique"`). Les classifieurs populaires en apprentissage automatique tels que les algorithmes naïfs de Bayes ou kNN ont été implémentés dans oMAP.

Finalement, nous avons également proposés un classifieur considérant la sémantique des définitions OWL en étant guidés par leur syntaxe. Ce classifieur structurel est décrit dans (Straccia et Troncy, 2005a). Il est utilisé dans notre outil *a posteriori* dans la mesure où l'ensemble des prédictions des autres classifieurs lui sert d'entrée et qu'il cherchera à étendre Σ en ajoutant des règles $T_j \leftarrow S_i$ si aucun appariement concernant T_j est déjà présent dans Σ.

Tous ces classifieurs ont donc été implémentés et sont compatibles avec l'API d'alignement décrite dans (Euzenat, 2004), ce qui facilite leur composition. Le problème d'aligner des ontologies a déjà été rapporté dans de nombreux travaux (Shvaiko et Euzenat, 2005). Cependant, il est difficile de comparer théoriquement les différentes approches. Depuis trois ans, l'Ontology Alignment Evaluation Initiative (OAEI[3]) propose des compétitions et des jeux de tests pour mesurer les forces et les faiblesses des différents outils. Nous avons évalué oMAP avec les données des campagnes d'évaluation de 2004 et 2005 (Straccia et Troncy, 2005b). Nous présentons brièvement dans la suite nos derniers résultats.

3 Résultats

oMAP est disponible à : `http://www.cwi.nl/~troncy/oMAP`.
Il doit être utilisé en lançant la commande :

```
java -jar omap.jar -i %method% -r %renderer%
     -o %resultFile% %sourceOnto% %targetOnto%
```

où :

[1] Par exemple en utilisant l'algorithme de Porter.
[2] WordNet® `http://wordnet.princeton.edu/`.
[3] `http://oaei.inrialpes.fr`

- *method* est : `it.cnr.isti.OMapAlignment` ;
- *renderer* is : `fr.inrialpes.exmo.align.impl.renderer.RDFRendererVisitor2` ;
- *resultFile* est le nom du fichier résultat ;
- *sourceOnto* et *targetOnto* sont les URIs absolus des ontologies source et cible à aligner.

Les résultats du test *directory* sont donnés dans le tableau 1 :

oMAP	**OLA**	**Falcon**	**Dublin20**	**CMS**	**FOAM**	**ctxMatch**
34.43%	31.96%	31.17%	26.53%	14.08%	11.88%	9.36%

TAB. 1 – *Résultats pour le test* directory*, classés selon les performances des participants, données 2005*

Les résultats du test *benchmark* sont donnés dans le tableau 2 :

#	Name	Prec.	Rec.	Time
101	Reference alignment	0.96	1.00	20ms
102	Irrelevant ontology	0.00	NaN	
103	Language generalization	0.96	1.00	20ms
104	Language restriction	0.96	1.00	20ms
201	No names	0.88	0.38	20ms
202	No names, no comments	0.85	0.24	20ms
203	No comments	0.96	1.00	
204	Naming conventions	0.95	0.89	40ms
205	Synonyms	0.81	0.63	40ms
206	Translation	0.89	0.49	40ms
207		0.89	0.49	40ms
208		0.96	0.90	30ms
209		0.73	0.54	40ms
210		0.90	0.39	40ms
221	No specialisation	0.96	1.00	20ms
222	Flatenned hierarchy	0.96	1.00	20ms
223	Expanded hierarchy	0.96	1.00	20ms
224	No instance	0.96	1.00	20ms
225	No restrictions	0.96	1.00	20ms
228	No properties	0.92	1.00	20ms
230	Flattened classes	0.91	1.00	20ms
231	Expanded classes	0.96	1.00	20ms
232		0.96	1.00	20ms
233		0.92	1.00	20ms
236		0.92	1.00	20ms
237		0.95	1.00	20ms
238		0.96	1.00	20ms
239		0.85	1.00	20ms
240		0.87	1.00	20ms
241		0.92	1.00	20ms
246		0.85	1.00	20ms
247		0.87	1.00	20ms
248		0.85	0.24	50ms
249		0.85	0.23	50ms
250		0.05	0.06	50ms
251		0.85	0.25	50ms
252		0.85	0.24	50ms
253		0.85	0.23	50ms
254		0.06	0.06	50ms
257		0.00	0.00	50ms
258		0.85	0.25	50ms
259		0.85	0.24	50ms
261		0.03	0.03	50ms
262		0.00	0.00	50ms
265		0.00	0.00	50ms
266		0.00	0.00	50ms
301	Real : BibTeX/MIT	0.94	0.25	40ms
302	Real : BibTeX/UMBC	1.00	0.58	40ms
303	Real : Karlsruhe	0.90	0.79	40ms
304	Real : INRIA	0.91	0.91	40ms

TAB. 2 – *Résultats détaillés de oMAP pour le test* benchmark

4 Conclusion

Au fur et à mesure que le web sémantique grossit, de plus en plus d'ontologies sont créées. Le développement d'outils automatiques pour aligner ces ontologies devient donc fondamental. Nous avons développé et décrit **oMAP**, un cadre formel pour aligner des ontologies. oMAP peut être vu comme une boite à outils utilisant différents classifieurs pour estimer la qualité des règles d'appariement. Nous avons implémenté et testé les différents classifieurs sur les données indépendantes fournis annuellement par les campagnes OAEI 2004 et 2005.

Comme travaux futurs, nous prévoyons d'introduire des classifieurs supplémentaires (utilisant d'autres ressources terminologiques ou d'autres mesures de distance tels que la distance KL) et de multiplier les tests en les combinant. Si l'ajout de nouveaux classifieurs dans oMAP est relativement trivial théoriquement, un problème subsiste lorsqu'il s'agit de trouver la combinaison optimale de toutes les prédictions obtenues. Nous avons proposé une liste de priorité entre les différents classifieurs, mais celle-ci est forcément dépendante du contenu des ontologies (leur degré de formalité, leur complexité, etc.). Par conséquent, nous planifions de multiplier les variantes et les tests afin d'améliorer globalement les performances d'oMAP.

Remerciements

A Umberto Straccia pour nos nombreuses discussions et à ses commentaires constructifs.

Références

Doan, A., J. Madhavan, R. Dhamankar, P. Domingos, et A. Halevy (2003). Learning to Match Ontologies on the Semantic Web. *The VLDB Journal 12*(4), 303–319.

Euzenat, J. (2004). An API for ontology alignment. In *3^{rd} International Semantic Web Conference (ISWC'04)*, Hiroshima, Japon, pp. 698–712.

Shvaiko, P. et J. Euzenat (2005). A Survey of Schema-based Matching Approaches. *Journal on Data Semantics (JoDS) 4*, 146–171.

Straccia, U. et R. Troncy (2005a). oMAP: Combining Classifiers for Aligning Automatically OWL Ontologies. In *6^{th} International Conference on Web Information Systems Engineering (WISE'05)*, New York City, New York, USA, pp. 133–147.

Straccia, U. et R. Troncy (2005b). oMAP: Results of the Ontology Alignment Contest. In *Workshop on Integrating Ontologies*, Banff, Canada, pp. 92–96.

Straccia, U. et R. Troncy (2006). Towards Distributed Information Retrieval in the Semantic Web: Query Reformulation Using the oMAP Framework. In *3^{rd} European Semantic Web Conference (ESWC'06)*, Budva, Montenegro, pp. 378–392.

Summary

This paper presents a method and a tool for automatically aligning OWL ontologies, a crucial step for achieving the interoperability of heterogeneous systems in the Semantic Web.

Different components are combined for finding suitable mapping candidates (together with their weights), and the set of rules with maximum matching probability is selected. Various classifiers, terminological, machine learning-based and a classifier using the structure and the semantics of the OWL ontologies are proposed in oMAP. Our method has been implemented and we provide some results of an evaluation from the 2005 Campaign tests of the Ontology Alignment Evaluation Initiative.