Módulos de Alinhamento para o *AgreementMakerLight*

Ricardo Ferreira Guimarães
Supervisor: Profª. Dr.ª Renata Wassermann
IME-USP, São Paulo

Introdução e Objetivo

Com o recente aumento de sistemas que utilizam ontologias para representar o domínio de conhecimento envolvido, também aumenta a necessidade de integrar ou trabalhar simultaneamente com mais de uma ontologia, como nos casos de sistemas multiagentes, integração de dados, composição de serviços web, sistemas *peer-to-peer*, e a própria realização da Web Semântica [2].
Entretanto, ao lidar com diferentes representações de domínios, evidenciam-se os problemas de heterogeneidade: diferenças linguísticas, conceituais, metodológicas, semânticas, entre outras; que dificultam a tarefa de relacionar ontologias. Uma das soluções para atacar este problema, consiste no uso de alinhadores de ontologia.

Definições

Baseadas nas definições de Euzenat e Shvaiko [2]:

- **Ontologia**: um documento escrito com linguagem ontológica (OWL, RDF, entre outras), contendo entidades tais como (vide os exemplos na Figura 1):
  - **Classes**: representam um conjunto de indivíduos do domínio. Como exemplos, temos `Veículo` e `Automóvel`.
  - **Indivíduos**: um único objeto do domínio. Temos um único indivíduo representado: `Carro-A`.
  - **Relações**: representa uma associação entre duas entidades, por exemplo, `anunciante`.
  - **Tipos**: especificam o domínio de um valor, entre os mais comuns temos: inteiros, cadeias de caracteres, URI e booleano.
  - **Valores**: valor atribuído à uma propriedade, como *2014*, poderia ser valor de `Ano`.
- **Mapeamento**: quíntupla $(id, e, e', n, \Re)$, tal que:
  - id é um identificador do mapeamento;
  - e é uma entidade de O e e' de O' (O e O', ontologias);
  - n é uma medida de confiança, que possibilite comparação entre dois alinhamentos quanto a este aspecto (para escolher o mais confiável, por exemplo);
  - $\Re$ uma relação dentre: equivalência ($\equiv$), especialização ($\subseteq$) ou disjunção($\perp$).
- **Alinhamento**: conjunto de mapeamentos entre duas ou mais ontologias (representado pelas setas cheias que ligam as duas árvores).
- **Algoritmo de alinhamento**: qualquer método usado para obter mapeamentos. Entre as técnicas utilizadas estão: distância de edição, análise do grafo e uso de tesauros e dicionários.
- **Estratégia de alinhamento**: conjunto de configurações e sequências de algoritmos pré-definidos.
- **Alinhador de ontologias**: programa que utiliza uma ou mais estratégias de alinhamento para emparelhar ontologias.

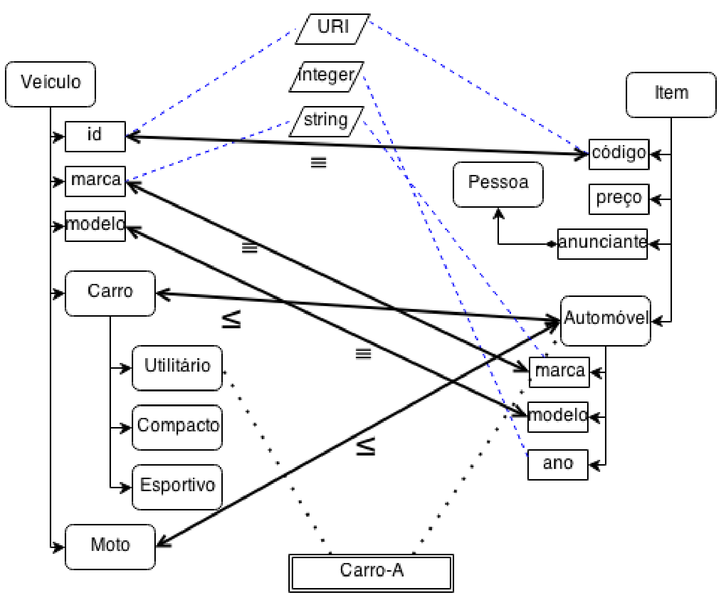

Figura 1: Representação gráfica de um trecho de duas ontologias e com alguns mapeamentos

AgreementMakerLight

Alinhador de código aberto (licença Apache 2.0), que possui resultados bastante promissores na OAEI (*Ontology Alignment Evaluation Initiative*) de 2013, principalmente, quando leva-se em consideração sua simplicidade, por não utilizar naquela competição nenhuma estratégia com base na estrutura da ontologia, nem ter tratamento para linguagens naturais múltiplas.

Arcabouço básico

Estruturas básicas:

- `Lexicon`: Concentra todas as informações léxicas obtidas, seja das próprias ontologias de entrada, ou de recursos externos.
- `RelationshipMap`: Em vez de guardar as relações entre classes na forma de grafos, o AML armazena todas as relações do tipo "é um" (equivalência) e "parte de" (especialização) com fecho transitivo, e as de disjunção sem fecho transitivo em uma tabela (junto com a distância da relação).
- `Mapping` e `Alignment`: Armazena-se cada mapeamento em um `Mapping`, contendo as classes envolvidas o tipo de mapeamento (qual a relação $\Re$) e a confiança. Um `Alignment` armazena e trata os mapeamentos obtidos ao longo de uma execução do alinhador

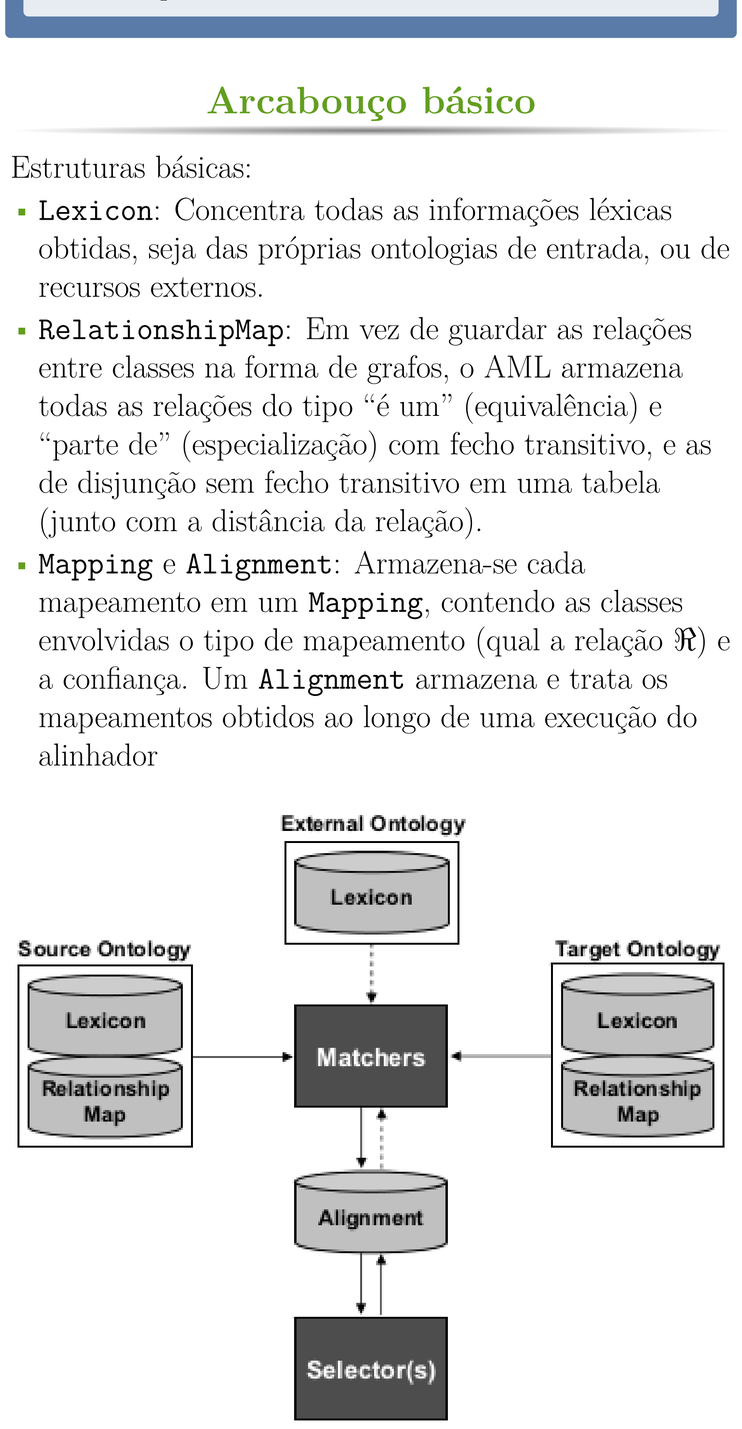

Figura 2: Esquema do AML, obtido de [4]

Uma das características do AML (*AgreementMakerLight*) consiste na sua modularização, o que permite fácil extensão de sua funcionalidade e definição de novas estratégias. Cada estratégia de alinhamento possui em seu núcleo, um conjunto de módulos de alinhamento (***matchers***) os quais consistem, basicamente, em algoritmos dedicados a encontrar mapeamentos (também podem incluir mais de um método e modificar estruturas que não sejam o `Alignment`).

Além disso, ele possui **seletores** capazes de remover correspondências pouco confiáveis e manter a cardinalidade dos mapeamentos em "um para um" (convertendo se necessário, em um conjunto equivalente).

A maior parte dos módulos faz análise de informações léxicas para descobrir mapeamentos entre as entidades. O AML também já incluía importantes módulos capazes de aproveitar recursos externos como o *WordNet* (tesauro da língua inglesa) outras ontologias como o *Uberon*.

Extras4AML

Para cumprir o objetivo de entender as diferentes soluções usadas no alinhamento, foram implementados 3 módulos com diferentes características de solução, em especial pensando nos problemas apontados em resultados da OAEI 2013 [3], os quais ainda estavam presentes na versão 2.05, a corrente quando criou-se o *fork* do projeto original.

Level 2 Jaro-Winkler

Para entender a diferença na variação de uma classe de algoritmos tão usadas (distância de edição), propõe-se utilizar uma alternativa simples, ainda não implementada, motivada pelos resultados em [1]. Usando como base a métrica original de Jaro-Winkler, calcula-se a similaridade entre duas cadeias de caracteres s e t da forma a seguir:

$$s = a_1 ... a_N \ (tokens)$$
$$t = b_1 ... b_M \ (tokens)$$
$$sim(s,t) = \frac{1}{N} \sum_{i=1}^{N} \max_{j=1}^{M} \text{Jaro-Winkler}(a_i, b_j)$$

DictionaryMatcher

Este módulo tem como objetivo principal extender o `Lexicon` com traduções obtidas a partir de um arquivo de dicionário. A ideia inicial era trabalhar com dados extraídos do *Wiktionary* usando o *wikt2dict* [5], entretanto, para que o dicionário seja lido basta apenas que o arquivo a ser esteja em um formato adequado.
Primeiramente, extendem-se os `Lexicons` das ontologias de entrada, itera-se sobre os pares de classes em ontologias distintas. Toma-se o conjunto de nomes de cada classe, e para cada língua, calcula-se a média das similaridades dos nomes, tendo como métrica o algoritmo `ISub`.

Descendant Analyser Matcher

Este módulo utiliza um conjunto de alinhamentos já existentes para inferir novos, observando os descendentes das classes. Para cada alinhamento, obtêm-se os conjuntos de "irmãos" (considerando a relação "é um") nas ontologias de entrada respectivas das classes mapeadas. A partir destes conjuntos, tomam-se os ascendentes em cada ontologia, e verifica-se para cada par destes, de ontologias distintas, a proporção dos filhos mapeados, e a quantidade de não correspondidos, para decidir se há uma relação entre pais ou não.

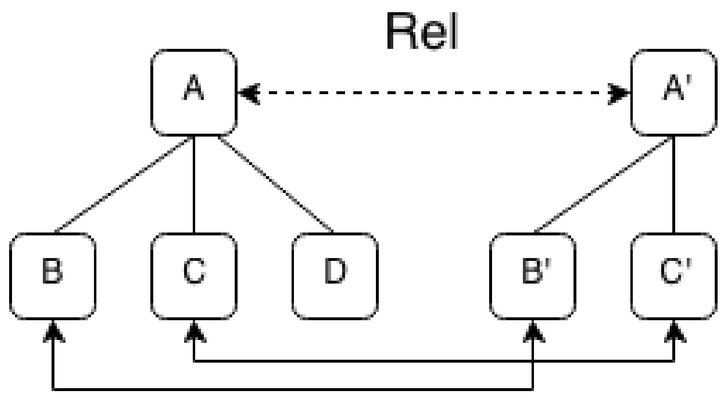

Figura 3: Ilustração do objetivo do *Descendant Analyser Matcher*

Referências

[1] William W. Cohen, Pradeep Ravikumar, and Stephen E. Fienberg. A comparison of string distance metrics for name-matching tasks. pages 73–78, 2003.
[2] Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer-Verlag, Heidelberg (DE), 2nd edition, 2007.
[3] Daniel Faria, Catia Pesquita, Emanuel Santos, Isabel F. Cruz, and Francisco M. Couto. AgreementMakerLight Results for OAEI 2013. URL http://disi.unitn.it/~p2p/OM-2013/oaei13_paper1.pdf.
[4] Daniel Faria, Catia Pesquita, Emanuel Santos, Matteo Palmonari, Isabel F. Cruz, and Francisco M. Couto. The AgreementMakerLight Ontology Matching System. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, volume 8185 of *Lecture Notes in Computer Science*, pages 527–541. Springer Berlin Heidelberg, 2013. doi: 10.1007/978-3-642-41030-7_38.
[5] Judit Ács. Pivot-based multilingual dictionary building using wiktionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may 2014.