



UNIVERSIDADE FEDERAL DE PERNAMBUCO

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CENTRO DE INFORMÁTICA

2011.1



**Explorando o Conhecimento Disponível na Web Semântica e o Feedback do
Usuário para Melhorar a Acurácia de Alinhamentos Gerados em Processos de
Ontology Matching**

DISSERTAÇÃO DE MESTRADO

Aluno: Thiago Pachêco Andrade Pereira (tpap@cin.ufpe.br)

Orientadora: Prof. Ana Carolina Salgado (acs@cin.ufpe.br)

Co-orientador: Prof. Carlos Eduardo Santos Pires (cesp@ufcg.dsc.edu.br)

Maio de 2011

UNIVERSIDADE FEDERAL DE PERNAMBUCO

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CENTRO DE INFORMÁTICA

2011.1

THIAGO PACHÊCO ANDRADE PEREIRA

**Explorando o Conhecimento Disponível na Web Semântica e o Feedback do
Usuário para Melhorar a Acurácia de Alinhamentos Gerados em Processos de
Ontology Matching.**

Trabalho apresentado à pós-graduação em
Ciência da Computação do Centro de
Informática da Universidade Federal de
Pernambuco.

Orientadora – Ana Carolina Salgado (acs@cin.ufpe.br)

Co-orientador – Carlos Eduardo Santos Pires
(cesp@ufcg.dsc.edu.br)

Recife, Maio de 2011

Dedico,

*A minha família, em especial
minha mãe e minha avó.*

Aos meus amigos.

© Thiago Pachêco Andrade Pereira

Todos direitos reservados.

Agradecimentos

Primeiramente agradecer a Deus por me proporcionar a oportunidade de realizar este trabalho. Depois a minha família pela paciência de perder feriados em casa devido aos estudos, principalmente a minha mãe e a minha avó.

Agradeço também a todos com quem estudei durante o período de cumprimento de créditos de cadeiras, em especial a Thiago Arruda, uma amizade mantida desde a graduação, que além de participar junto comigo do projeto SPEED, fez parte de vários projetos das cadeiras cursadas.

Um agradecimento especial a todos que fazem o projeto SPEED, por ter tornado o desenvolvimento deste trabalho mais prazeroso, incluindo Andrêza, que me ajudou muito na colaboração com o trabalho de Graduação de Thiago Henrique.

E por último, um agradecimento a minha orientadora Carol, e ao meu co-orientador Carlos, pela paciência e pelo apoio, mesmo com as complicações de tempo por eu estar trabalhando. Não sei onde conseguiria outros orientadores que me dessem tanto apoio quanto foi me dado e por isso agradeço.

Resumo

Ontologias são usadas em diversos tipos de aplicações, como integração de banco de dados, sistemas *peer-to-peer*, *e-commerce*, serviços na Web Semântica e redes sociais. Devido ao aumento na quantidade e diversidade de ontologias, surge o problema da heterogeneidade de termos e representações e um dos desafios do uso de ontologias é tratar de forma mais eficiente essa heterogeneidade para possibilitar um melhor entendimento entre sistemas e a integração de informações.

Um processo de *ontology matching* tem como objetivo encontrar correspondências entre elementos de ontologias distintas. Isto faz com que ferramentas deste tipo sejam interessantes em aplicações que usam ontologias. O conjunto de correspondências gerado é chamado alinhamento e pode ser usado para criar um entendimento entre agentes que representam seu conhecimento por ontologias, como também para reformular consultas em bases de dados que possuam seus esquemas representados por ontologias, possibilitando a integração de dados, dentre outras aplicações. Muitas técnicas de *ontology matching* usam comparações sintáticas ou estruturais entre os elementos das próprias ontologias. Porém, usar apenas o conhecimento interno das ontologias não tem sido suficiente para inferir todas as correspondências possíveis entre as ontologias envolvidas.

Este trabalho tem como objetivo o uso de novas fontes de conhecimento externo encontradas na Web Semântica. Ontologias são usadas para adicionar semântica à navegação na Web, sendo associadas a páginas e contendo informações semânticas sobre elas. O desenvolvimento da Web Semântica vem fazendo crescer o número de ontologias disponíveis na Internet. Usamos estas ontologias para extrair conhecimento adicional sobre um determinado domínio do qual faz parte as ontologias que são comparadas.

Outro objetivo deste trabalho é fornecer ao usuário a possibilidade de remover correspondências para obter conhecimento sobre correspondências incorretas. Essas informações são guardadas e, em futuras operações de *ontology matching* que envolvam ontologias de domínio semelhante, podem ser levadas em consideração para incluir ou eliminar uma correspondência de um alinhamento futuro.

Palavras-chave: Comparação de ontologias, Web Semântica, Feedback do usuário, Ontologia de Domínio

Abstract

Ontologies are used in many kinds of applications such as database integration, peer-to-peer systems, e-commerce, Web Semantic services, and social networks. The diversity and the quantity of ontologies arise the problem of heterogeneity of terms and domain representations. One of the challenges is to decrease this heterogeneity to make possible systems communication and information integration.

An ontology matching process aims at finding correspondences between elements in different ontologies. This fact makes this kind of tool interesting in applications that use ontologies. The set of correspondences generated is called alignment and can be used to create an understanding among agents that represents its knowledge using ontologies. It can also be used to reformulate queries over databases that represent its schemas using ontologies allowing, for instance, the data integration. Many techniques use syntactic and structural comparison between ontology elements. However, the knowledge available inside the ontology itself is not enough, provoking the investigation of new sources of knowledge.

One of the goals of this work is to use external knowledge available on the Semantic Web. Ontologies are used to add semantics to Internet navigation, they are associated to sites and contain semantic information about them. The growing of the Semantic Web has increased the number of ontologies available in the Internet. These ontologies can be used to find additional knowledge about a certain domain in which the compared ontologies are part of.

Another goal of this work is to explore the knowledge of the user feedback. Providing to the users the possibility of removing correspondences is a good way to obtain knowledge about incorrect correspondences. This information is stored and can be used in future executions of the matching operation. They can also be used to decide if the correspondence will be included or not in the final alignment.

Keywords: ontology matching, Semantic web, user feedback, domain ontologies

Sumário

1. Introdução	1
1.1 Motivação.....	1
1.2 Definição do Problema	2
1.3 Objetivo	3
1.4 Estrutura do Documento.....	4
2. Conceitos e Técnicas Usadas em <i>Ontology Matching</i>	6
2.1 Definições.....	6
2.2 <i>Ontology Matching</i>	7
2.3 Técnicas de <i>Ontology Matching</i> Existentes.....	8
2.3.1 Técnicas que Trabalham no Nível de Elementos.....	9
2.3.2 Técnicas que Trabalham no Nível de Estrutura.....	11
2.3.3 Técnicas Extensionais	11
2.3.4 Técnicas Semânticas.....	11
2.4 Usos de <i>Matching</i> de Ontologias	15
2.5 Métodos para Avaliação de Alinhamentos	16
2.5.1 Medidas de Conformidade.....	16
2.5.2 Medidas de Desempenho	18
2.6 Considerações Finais	18
3. Ferramentas de <i>Ontology Matching</i> Existentes	19
3.1 CUPID.....	19
3.2 COMA++	20
3.3 C-SAW	21
3.4 S-Match	21
3.5 ctxMatcher	22
3.6 SCARLET Matcher	22
3.7 <i>GeRoMeSuite</i>	23
3.8 ASMOV	23
3.9 <i>SemMatcher</i>	24
3.10 Considerações Finais	30
4. <i>SemMatcher+</i>	32
4.1 Arquitetura do <i>SemMatcher+</i>	32
4.2 <i>Matcher</i> Linguístico-Estrutural.....	35

4.3	Execução do <i>Matcher</i> Semântico	36
4.3.1	Regras de Propriedades	36
4.3.2	Gerenciador de Sinônimos	39
4.3.3	Processos do <i>Matcher</i> Semântico	39
4.4	Histórico	43
4.5	<i>Feedback</i> do Usuário	45
4.6	Considerações Finais	46
5.	Implementação e Experimentos	47
5.1	Tecnologias utilizadas.....	47
5.1.1	Bibliotecas para o <i>WordNet</i>	47
5.1.2	Engenhos de busca de ontologias	48
5.1.3	Controle de histórico das correspondências.....	49
5.2	Experimentos	51
5.2.1	Uso de Sinônimos nas Regras Semânticas	51
5.2.2	Conhecimento Externo.....	52
5.2.3	<i>Feedback</i> do Usuário	55
5.3	Considerações Finais	58
6.	Conclusão e Trabalhos Futuros.....	60
6.1	Contribuições	60
6.2	Trabalhos Futuros.....	61
	Referências	64
	Apêndice A	70
	Apêndice B	88
	Apêndice C	100
	Apêndice D	116

Lista de Figuras

Figura 2.1 - Classificação de técnicas de *ontology matching* [Euzenat and Shvaiko 2007a]

Figura 2.2 – Inferência de relacionamentos consultando ontologia externa [Sabou *et al.* 2008].

Figura 2.3 – Inferência de correspondência consultando apenas uma ontologia [Sabou *et al.* 2008].

Figura 2.4 – Inferência de correspondências consultando mais de uma ontologia [Sabou *et al.* 2008].

Figura 2.5 – Exemplo de desambiguação feita usando análise do contexto do conceito.

Figura 3.1 – Processo de *matching* do CUPID [Euzenat and Shvaiko 2007a]

Figura 3.2 – Arquitetura do COMA++ [AumueLLer *et al.* 2005]

Figura 3.3 – Arquitetura da primeira versão do *SemMatcher* [Pires *et al.* 2009]

Figura 3.4 – Correspondência do tipo *isEquivalentTo* [Souza 2009]

Figura 3.5 – Correspondência do tipo *isSubConceptOf* [Souza 2009]

Figura 3.6 – Correspondência do tipo *isSuperConceptOf* [Souza 2009]

Figura 3.7 – Correspondência do tipo *isPartOf* [Souza 2009]

Figura 3.8 – Correspondência do tipo *isWholeOf* [Souza 2009]

Figura 3.9 – Correspondência do tipo *isCloseTo* [Souza 2009]

Figura 3.10 – Correspondência do tipo *isDisjointWith* [Souza 2009]

Figura 3.11 – Medidas disponíveis no *SemMatcher* [Pires *et al.* 2009]

Figura 4.1 – Nova arquitetura do *SemMatcher*

Figura 4.2 - Ontologias usadas para descrever o processo do *SemMatcher+*

Figura 4.3 - Relacionamento de equivalência entre propriedades.

Figura 4.4 – Relacionamento de especialização entre propriedades.

Figura 4.5 – Relacionamento de generalização entre propriedades.

Figura 4.6 – Processo do *matcher* semântico

Figura 4.7 – Identificação de processos de *matcher* anteriores semelhantes ao atual.

Figura 5.1 – Arquitetura do *Watson* [d’Aquin *et al.* 2007]

Figura 5.2 – Esquema relacional para o histórico de correspondências

Figura 5.3 – Gráfico mostrando a evolução da precisão e cobertura nas versões do *SemMatcher* com e sem o uso do *WordNet*

Figura 5.4 – Gráfico mostrando a evolução da precisão e cobertura do *SemMatcher* no domínio de educação com e sem o uso de conhecimento externo

Figura 5.5 – Gráfico mostrando a evolução da precisão e cobertura do *SemMatcher* no domínio de animais com e sem o uso de conhecimento externo

Figura 5.6 – Tela com resultado final para avaliação do usuário.

Figura 5.7 – Tela de avaliação de resultados do SPEED [Paz 2010]

Figura A.1 – Ontologia de entrada 1 do domínio de animais do experimento de uso *WordNet*

Figura A.2 – Ontologia de entrada 2 do domínio de animais do experimento de uso *WordNet*

Figura A.3 – Primeira parte da ontologia de domínio usada no experimento de uso *WordNet*

Figura A.4 – Segunda parte da ontologia de domínio usada no experimento de uso *WordNet*

Figura A.5 – Terceira parte da ontologia de domínio usada no experimento de uso *WordNet*

Figura B.1 – Ontologia de entrada 1 do primeiro experimento de conhecimento externo usando o domínio de educação

Figura B.2 – Ontologia de entrada 2 do primeiro experimento de conhecimento externo usando o domínio de educação

Figura B.3 – Ontologia de domínio do primeiro experimento de conhecimento externo usando o domínio de educação

Figura C.1 – Ontologia de entrada 1 do segundo experimento de conhecimento externo usando o domínio de animais

Figura C.2 – Ontologia de entrada 2 do segundo experimento de conhecimento externo usando o domínio de animais

Figura C.3 – Primeira parte da ontologia de domínio usada no segundo experimento de conhecimento externo sobre o domínio de animais

Figura C.4 – Segunda parte da ontologia de domínio usada no segundo experimento de conhecimento externo sobre o domínio de animais

Figura C.5 – Terceira parte da ontologia de domínio usada no segundo experimento de conhecimento externo sobre o domínio de animais

Figura D.1 – Ontologia 1 do cenário de Educação para o experimento de histórico

Figura D.2 – Ontologia 2 do cenário de Educação para o experimento de histórico

Figura D.3 – Ontologia 3 do cenário de Educação para o experimento de histórico

Figura D.4 – Ontologia 4 do cenário de Educação para o experimento de histórico

Figura D.5 – Ontologia 5 do cenário de Educação para o experimento de histórico

Figura D.6 – Ontologia 6 do cenário de Educação para o experimento de histórico

Figura D.7 – Ontologia 7 do cenário de Educação para o experimento de histórico

Figura D.8 – Ontologia 8 do cenário de Educação para o experimento de histórico

Figura D.9 – Ontologia 9 do cenário de Educação para o experimento de histórico

Figura D.10 – Ontologia 10 do cenário de Educação para o experimento de histórico

Figura D.11 – Ontologia 11 do cenário de Educação para o experimento de histórico

Figura D.12 – Ontologia 12 do cenário de Educação para o experimento de histórico

Figura D.13 – Primeira parte da ontologia de domínio do cenário de Educação para o experimento de histórico

Figura D.14 – Segunda parte da ontologia de domínio do cenário de Educação para o experimento de histórico

Figura D.15 – Terceira parte da ontologia de domínio do cenário de Educação para o experimento de histórico

Lista de Quadros

Quadro 3.1 - Comparação entre as ferramentas de *ontology matching* analisadas e o *SemMatcher*.

Quadro 4.1 - Alinhamento A_{LE} gerado.

Quadro 4.2 - Alinhamento A_{S1} gerado.

Quadro 4.3 - Alinhamento A_{S2} gerado.

Quadro 4.4 - Alinhamento A_S gerado.

Quadro 4.5 - Alinhamento A_F gerado.

Quadro 5.1 – Comparativo de tempo e número de correspondências com e sem o uso do *WordNet*

Quadro 5.2 – Comparativo de tempo e número de correspondências do *SemMatcher* no cenário de educação com e sem o uso de conhecimento externo

Quadro 5.3 – Comparativo de tempo e número de correspondências do *SemMatcher* no domínio de animais com e sem o uso de conhecimento externo

Quadro 5.4 - Atualização do grau de confiança durante as execuções

Glossário de Siglas

API	Application Programming Interface
DAML	DARPA Agent Markup Language
NLP	Natural Language Processing
OAEI	Ontology Alignment Evaluation Initiative
OPDMS	Ontology-based Peer Data Management Systems
OWL	Web Ontology Language
P2P	Peer-to-peer
PDMS	Peer Data Management System
RAM	Random Access Memory
RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
URI	Uniform Resource Identifier
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

1. Introdução

Neste capítulo, apresentamos a motivação para este trabalho, o contexto do problema a ser resolvido, e os objetivos gerais e específicos, além da estrutura do documento.

1.1 Motivação

Uma ontologia é uma especificação de uma conceitualização. O termo tem origem na filosofia onde uma ontologia é uma sistemática da existência. Para sistemas baseados em conhecimento, a existência é o que pode ser representado [Gruber 1993]. Ontologias têm sido usadas como uma forma de uniformizar o compartilhamento de informações em diversos tipos de aplicações, descrevendo um domínio de interesse, seja representando um esquema de banco de dados, uma classificação de conceitos ou axiomas de um teorema. Com o uso de ontologias, todo esse conhecimento é representado de uma forma que possa ser interpretado por um computador [Fensel 2004]. Muitas vezes, o conhecimento expresso em uma ontologia não é completo, sendo necessário efetuar operações sobre ontologias para conseguir conhecimento adicional. Uma dessas operações é de comparação de ontologias, chamada de *ontology matching*, que possibilita encontrar semelhanças e novos relacionamentos entre ontologias [Euzenat and Shvaiko 2007a].

Na Web Semântica [Berners-Lee *et al.* 2001], ontologias são usadas para facilitar a navegação na Internet. Ontologias são associadas aos sites, e possuem informações sobre a semântica do conteúdo dos sites. *Browsers* como o *Magpie* [Dzbor *et al.* 2004] foram desenvolvidos aproveitando esse conhecimento para prover páginas semelhantes (usando *matching* entre as ontologias dos sites) ou com conteúdo mais detalhado. Além disso, informações adicionais como instâncias das ontologias, são destacadas no conteúdo do site e *links* para dados sobre as instâncias são criados.

O *browser SemWeb* [Sah *et al.* 2008], além de prover as mesmas facilidades que o *Magpie*, faz uso de personalização para melhorar ainda mais a experiência para cada usuário. Ao contrário do *Magpie*, o *SemWeb* foi implementado como um *plugin* para o *browser Firefox*. Ainda no campo da Web Semântica, ontologias facilitam a busca de conteúdo dentro de um site e provêem informações adicionais para a indexação de sites.

Ontologias também são usadas na comunicação entre agentes autônomos [Bouquet *et al.* 2003], onde funcionam como vocabulário comum entre os agentes, e cada agente possui a sua ontologia com o seu conhecimento sobre o mundo. Para haver

entendimento na comunicação entre os agentes, operações de *matching* são efetuadas entre as ontologias dos agentes que estão se comunicando para comparar o conhecimento de cada um e facilitar a comunicação, encontrando relacionamentos entre os conceitos presentes nas ontologias dos mesmos.

No campo da integração de informações [Halevy *et al.* 2005], ontologias são usadas como uma forma de uniformizar a representação de esquemas de bancos de dados. Utilizando *ontology matching*, é possível fazer integração de esquemas e de dados, identificando entidades semanticamente semelhantes.

Dentre os sistemas *peer-to-peer* (P2P), destacamos os *Peer Data Management Systems* (PDMS) [Sung *et al.* 2006], que permitem acesso de forma transparente a bases de dados heterogêneas e distribuídas, sem a necessidade de um controle centralizado. Em outra abordagem, Xiao [Xiao 2006] propôs o uso de ontologias para representação dos esquemas de banco de dados, criando assim o conceito de *Ontology-based Peer Data Management System* (OPDMS), surgindo a necessidade de operações sobre ontologias para reformulação de consultas e identificação de *peers* semanticamente semelhantes na rede.

Como foi mostrado, em todos os cenários apresentados, existe a necessidade de descobrir relacionamentos e semelhanças entre esquemas e representações de conhecimento. Nesse sentido, o *SemMatcher* [Pereira 2008] é uma ferramenta de comparação de ontologias (*ontology matching*) usada no sistema SPEED (*Semantic Peer Data Management System*) [Pires *et al.* 2006]. O SPEED é um OPDMS em que cada *peer* é uma fonte de dados autônoma e disponibiliza seu esquema local usando uma ontologia. O resultado do *SemMatcher* é um alinhamento que pode ser usado para (i) agrupar semanticamente os *peers* na rede: neste caso, uma medida de similaridade global entre os *peers* é calculada e usada para agrupá-los; e (ii) reformular consultas: quando uma consulta é submetida a um *peer*, ela é reformulada e enviada para os *peers* vizinhos usando as correspondências identificadas entre as ontologias dos *peers*.

1.2 Definição do Problema

Efetuar uma operação de *ontology matching* tem como objetivo descobrir similaridades entre as ontologias comparadas, inferindo relacionamentos entre os seus elementos. São quatro os tipos de heterogeneidade descritos por [Euzenat and Shvaiko 2007a]:

- Sintática: este tipo de heterogeneidade se dá quando as linguagens para representação de ontologias são diferentes. Por exemplo, quando uma ontologia é

representada em OWL (*Web Ontology Language*) [OWL 2011], e outra em DAML (*DARPA Agent Markup Language*) [DAML 2011];

- Terminológica: é o tipo de heterogeneidade no qual diferentes palavras ou idiomas são usados para representar um conceito ou propriedade. Este tipo de heterogeneidade pode acontecer devido ao fato de duas ontologias serem escritas em idiomas diferentes (por exemplo, *Artigo* x *Article*), ou pelo uso de sinônimos (por exemplo, *Aluno* x *Estudante*);

- Conceitual ou semântica: está presente tanto na cobertura de conceitos de cada ontologia, quanto na granularidade da hierarquia de níveis ou ainda na perspectiva ou visão do domínio que a ontologia representa. Por exemplo, duas ontologias que descrevem uma mesma região do planeta podem ter diferentes perspectivas. Uma focando nos aspectos políticos e outra nos aspectos geológicos;

- Semiótica: quando há diferença de contexto do significado de um conceito [Bouquet *et al.* 2004]. Diferentes pessoas ou comunidades podem interpretar de uma forma diferente uma mesma ontologia. Por exemplo, se o conceito *Europa* aparece em um esquema de classificação de um repositório multimídia com uma hierarquia semelhante a *Imagens* → *P&B* → *Europa*, não deve ser considerado equivalente a um conceito *Europa* localizado em uma ontologia geográfica [Bouquet *et al.* 2005].

Grande parte dos erros na descoberta de correspondências tem a ver com o entendimento do domínio ou do contexto em que aquela ontologia está sendo usada, ou seja, a semântica. Por isso, dentre os novos desafios no campo de *ontology matching* apontados por [Shvaiko and Euzenat 2008], destacamos o envolvimento do usuário no processo de *matching*, e a descoberta de novo conhecimento com o uso de *background knowledge* (conhecimento pré-existente sobre o domínio, que pode estar contido em uma outra ontologia, um *thesaurus* ou referir-se a qualquer outra forma de obter um conhecimento adicional).

1.3 Objetivo

O objetivo geral deste trabalho é analisar as soluções de *ontology matching* que consideram a semântica no seu processo [Tierney and Jackson 2008] [Giunchiglia *et al.* 2009; Sabou *et al.* 2007], e propor mudanças no processo de *ontology matching* atual da ferramenta *SemMatcher*.

Como objetivos específicos ressaltamos buscar mecanismos para:

- a) melhorar a geração de correspondências, possibilitando a descoberta de correspondências não encontradas pelo atual processo do *SemMatcher*. Para

suprir essa necessidade, analisamos técnicas que fazem uso da Web Semântica para buscar novas ontologias que podem conter esse conhecimento. A Web Semântica tem se tornado cada vez mais uma realidade e o número de ontologias na Internet tem aumentado devido a esse fato [Finin 2005]. Propomos o uso do conhecimento disponível nessas ontologias para enriquecer os alinhamentos gerados;

- b) permitir a remoção de correspondências incorretas que sejam encontradas tanto na execução do processo normal, como na obtenção de ontologias externas. Para esse fim, permitimos ao usuário avaliar as correspondências geradas ao final do processo. Em futuras execuções do *SemMatcher*, correspondências que têm avaliações negativas poderão ser eliminadas, dependendo do quanto são consideradas confiáveis pela ferramenta;
- c) melhorar a inferência dos relacionamentos no *matcher* semântico. As inferências são feitas com o auxílio de uma ontologia de domínio para a descoberta de relacionamentos semânticos. Uma das formas de melhorar sua inferência é consultando sinônimos no *WordNet* [Fellbaum 1998], adicionando semântica na inferência de relacionamentos de equivalência ao comparar não apenas os termos das ontologias que possuem grafia semelhante, como também seus sinônimos;
- d) melhorar a inferência de relacionamentos entre propriedades da ferramenta *SemMatcher*. Não apenas a propriedade é usada para comparação, mas o conceito que possui a propriedade também é analisado para inferir os relacionamentos de equivalência, subpropriedade e superpropriedade.

1.4 Estrutura do Documento

Os demais capítulos deste documento seguem a seguinte organização:

- Capítulo 2: apresenta conceitos básicos relacionados a ontologias como o uso da linguagem OWL, as operações sobre ontologias, principalmente o *matching* de ontologias. Cada uma das técnicas é detalhada e apresentada de acordo com sua classificação. Também são mostradas aplicações de *ontology matching* em sistemas e, por último, as medidas de acurácia de alinhamento definidas pela *Ontology Alignment Evaluation Initiative* (OAEI) [OAEI 2011];
- Capítulo 3: descreve ferramentas de *ontology matching* existentes associando cada uma das técnicas descritas no Capítulo 2. O capítulo descreve ainda a

ferramenta *SemMatcher* em sua primeira versão. Ao final do capítulo um resumo comparativo entre as ferramentas analisadas é feito;

- Capítulo 4: analisa de forma mais aprofundada as técnicas propostas a serem usadas na nova versão do *SemMatcher*. Além disso, detalha a nova arquitetura e as etapas de funcionamento do novo processo do *SemMatcher*;
- Capítulo 5: descreve os experimentos feitos para avaliar o uso de cada técnica proposta e os resultados obtidos comparando-os com os resultados gerados com a versão anterior do *SemMatcher*;
- Capítulo 6: descreve as conclusões e dificuldades encontradas durante o desenvolvimento deste trabalho. Também são citados trabalhos futuros a serem realizados com a ferramenta.

2. Conceitos e Técnicas Usadas em *Ontology Matching*

Alguns conceitos importantes são necessários para uma melhor compreensão deste trabalho. Começamos por definições sobre o processo de *ontology matching*, depois explicamos o seu funcionamento e, por fim, analisamos técnicas que usam semântica ou contexto em seu processo de descoberta de correspondências entre elementos de ontologias distintas.

2.1 Definições

Ontologia é um conceito desenvolvido pela comunidade de Inteligência Artificial com a intenção de facilitar o compartilhamento e reuso de conhecimento [Guarino 1998]. Ontologias visam descrever um domínio específico que se deseja modelar. O resultado dessa modelagem é um grafo na forma de uma hierarquia de conceitos. Cada conceito pode ter propriedades, que são atributos do conceito. Além da parte conceitual, ontologias podem conter instâncias.

A linguagem de representação de ontologias padrão da W3C [W3C 2011] é chamada de OWL (*Web Ontology Language*) [OWL 2011], e é uma extensão da linguagem chamada RDF (*Resource Description Framework*) [RDF 2011]. A linguagem OWL é escrita em XML, porém é usada para representar os relacionamentos entre as entidades dentro de uma ontologia. Esses relacionamentos são expressos por axiomas, que são uma tripla formada por sujeito, predicado e objeto, onde geralmente o sujeito e o objeto são conceitos ou propriedades, e o predicado é o tipo de relacionamento entre o sujeito e o objeto.

A definição de OWL como padrão W3C [W3C 2011] para representação de ontologias contribuiu para o compartilhamento de conhecimento expresso por ontologias. A construção de uma ontologia única contendo todo conhecimento do mundo é impraticável, até mesmo sobre um único domínio pode ser impraticável criar uma ontologia única com todo conhecimento disponível, o que levou ao surgimento de uma diversidade de ontologias criadas sobre domínios específicos. Para unir o conhecimento entre as diversas ontologias foram definidas operações entre ontologias. Uma das mais importantes é a comparação de ontologias, que vamos detalhar na próxima seção.

2.2 *Ontology Matching*

Ontology Matching tem como objetivo encontrar correspondências entre entidades semanticamente relacionadas de diferentes ontologias. Chamamos de *matcher* as ferramentas ou componentes que executem *ontology matching*, e chamamos de *matching*, o processo em si. O resultado de um *ontology matching* é um conjunto de correspondências entre entidades chamado alinhamento. Segundo [Euzenat and Shvaiko 2007b] um alinhamento entre duas ontologias o e o' é um conjunto de correspondências que podem ser representadas por uma 4-tupla (e, e', r, w) , onde e é um elemento que pertence a o (podendo ser um conceito ou propriedade) e é chamado de sujeito, e' pertence a o' (também podendo ser um conceito ou propriedade) e é chamado de objeto, r é o tipo de relacionamento entre e e e' , também chamado de predicado, e w é o peso da correspondência. Dentre os tipos de relacionamentos possíveis estão:

- Equivalência: quando dois conceitos ou propriedades são equivalentes;
- Subconceito: quando um conceito ou propriedade é um nó filho de outro conceito ou propriedade na hierarquia do grafo formado pela ontologia;
- Superconceito: quando um conceito ou propriedade é um nó pai de outro conceito ou propriedade na hierarquia do grafo formado pela ontologia;
- Disjunção: quando dois conceitos não têm nenhuma semelhança.

É comum confundir *ontology matching* com *schema matching*. Segundo [Euzenat and Shvaiko 2007a], *ontology matching* explora conhecimento explícito expresso em ontologias, e *schema matching* explora o significado expresso em esquemas de dados (podendo ou não ser uma ontologia). Estes dois tipos de processo usam técnicas similares que podem ser aplicadas em ambos cenários. A arquitetura geral para uma operação de *schema matching* descrita em [Rahm and Bernstein 2001] mostra que, para fazer o *matching* entre esquemas, eles devem ser transformados em uma representação interna, que é usada pelo *matching* para encontrar as correspondências. Em ferramentas de *schema matching*, essa representação interna pode ser feita por ontologias, neste caso, tornando as técnicas usadas em *ontology* e *schema matching* ainda mais parecidas.

Os alinhamentos gerados podem ser úteis para integrar informações e instâncias das duas ontologias, identificar a semelhança entre elas verificando a quantidade de conceitos em comum, ou mesmo servir como entrada para uma operação de integração de ontologias (*ontology merging*). Um processo de *merging* tem como objetivo gerar uma ontologia única coerente a partir de duas ou mais ontologias diferentes que

descrevem um mesmo assunto [Pinto *et al.* 1999]. Os alinhamentos encontrados podem ter as seguintes cardinalidades:

- **1:1:** para cada elemento da primeira ontologia, há uma correspondência em relação a apenas outro elemento da segunda ontologia, que pode ser a correspondência de maior similaridade entre as correspondências geradas com cardinalidade 1:N;
- **1:N:** para cada elemento da primeira ontologia, há uma correspondência com valor de similaridade para mais de um elemento na segunda ontologia. Neste ponto, é possível fazer uso de um limiar para eliminar correspondências que tenham valores de similaridade muito baixos;
- **N:1:** semelhante à cardinalidade anterior mas acontece no sentido da segunda ontologia para a primeira;
- **N:M:** tanto elementos da primeira ontologia possuem correspondências com vários elementos da segunda ontologia como os da segunda ontologia possuem correspondências com vários elementos da primeira ontologia.

Para a identificação de correspondências, um processo de *ontology matching* pode usar um ou vários tipos de técnicas sobre os conceitos e propriedades das ontologias de entrada. Algumas dessas técnicas são descritas na próxima seção.

2.3 Técnicas de *Ontology Matching* Existentes

Diferentes técnicas têm sido usadas pelas ferramentas de *matching* para gerar alinhamentos. Na classificação apresentada por [Euzenat and Shvaiko 2007a], primeiro temos dois grandes grupos: os algoritmos que exploram a estrutura e os que exploram as instâncias. Dentro de cada grupo, existem técnicas específicas como vemos na Figura 2.1.

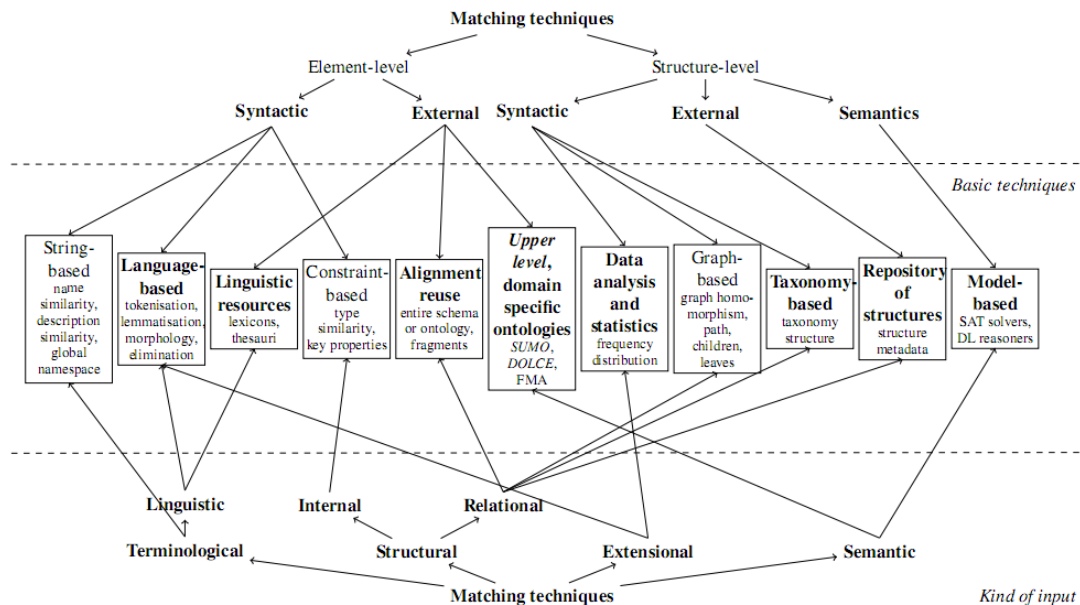


Figura 2.1 – Classificação de técnicas de *ontology matching* [Euzenat and Shvaiko 2007a]

As técnicas que trabalham no nível de elemento comparam conceitos isoladamente, sem levar em consideração seus relacionamentos com outros conceitos na mesma ontologia. Já as técnicas estruturais comparam conceitos explorando os relacionamentos já existentes com conceitos da ontologia que pertence. Nas próximas seções, analisaremos os principais algoritmos referentes a esses dois tipos de técnicas.

2.3.1 Técnicas que Trabalham no Nível de Elementos

As ferramentas de *ontology matching* mais antigas faziam uso de técnicas que se baseavam puramente em comparações de *string*, usando recursos como dicionário de radicais e cálculos de distância, ou analisando a estrutura de conceitos [Aumuellner *et al.* 2005] [Euzenat 2004], sem considerar a semântica dos termos da ontologia.

Técnicas Baseadas na Sintaxe

As técnicas baseadas em sintaxe usam o nome dos conceitos e de suas propriedades, além das URIs (*Uniform Resource Identifiers*) das ontologias para a sua comparação. Em [Euzenat and Shvaiko 2007a], há uma classificação das técnicas existentes, onde são definidos os seguintes tipos:

- Baseadas em *string*: consideram a estrutura da *string* do rótulo do conceito durante a comparação. São basicamente quatro as técnicas usadas:
 - Normalização: antes de comparar a *string* que representa os conceitos e propriedades, normaliza seus nomes, sobrando apenas o radical da palavra. A comparação entre conceitos e propriedades é feita pelo radical.

- *Substring*: supõe que, se uma *string* que representa um conceito está contida dentro da *string* de outro conceito, eles têm uma similaridade maior.
- *Edit distance*: para comparar dois elementos, é calculado o número de alterações necessárias para chegar à outra *string*. Quanto maior o número de alterações, maior a distância entre as *strings*.
- *Token-based*: antes de comparar os elementos, se a palavra é composta, ela é dividida e cada palavra é comparada separadamente para verificar se são semelhantes.
- Baseadas na linguagem: Estas técnicas usam Processamento de Linguagem Natural (NLP – *Natural Language Processing*) para extrair termos significativos das descrições dos elementos das duas ontologias a serem comparadas. Entre elas estão técnicas comumente conhecidas em engenhos de busca como: retirada de *stopwords*, identificação de *stemming*, dentre outras.

Técnicas Baseadas em Conhecimento Externo

Essas técnicas buscam conhecimento adicional fora das ontologias que estão sendo comparadas. Esse conhecimento externo pode vir de um *thesaurus*, de uma ontologia de domínio ou de um dicionário genérico como o *WordNet* [Fellbaum 1998]. Duas são as formas de obter conhecimento externo existentes:

- Recursos linguísticos externos: Usam de alguma forma dicionário de radicais, *thesaurus* e *WordNet* para comparar conceitos. Dois conceitos são considerados equivalentes, mesmo com a grafia totalmente diferente, se for descoberto em algum desses recursos externos que eles são sinônimos.
- Reuso de alinhamentos: O reuso de alinhamentos é motivado pela intuição de que muitas ontologias que passarão pelo processo de *matching* são similares a ontologias que já passaram pelo mesmo processo, principalmente se representam o mesmo domínio [Euzenat and Shvaiko 2007a]. Se em um processo anterior foi armazenado uma correspondência entre dois conceitos, e em um processo futuro é necessário descobrir o relacionamento entre os mesmos dois conceitos, reusa-se o relacionamento anterior.

2.3.2 Técnicas que Trabalham no Nível de Estrutura

As técnicas baseadas na estrutura das ontologias analisam a estrutura interna dos conceitos (as propriedades) para verificar o quão semelhante eles são, assim como suas distâncias no grafo que representa a ontologia. São basicamente de dois tipos:

- Interna: como o nome sugere, analisa a estrutura interna dos conceitos (as propriedades) e as compara. Esta técnica é particularmente usada em *schema matching* [Rahm and Bernstein 2001].
- Relacional: percorre o grafo da ontologia para ver a distância entre os dois conceitos que estão sendo comparados. Este problema é conhecido como o problema do homomorfismo de grafos [Garey and Johnson 1979]

2.3.3 Técnicas Extensionais

Para calcular a similaridade entre conceitos nas ontologias de entrada, esta técnica faz comparações entre as instâncias. Quando dois conceitos têm instâncias semelhantes fica mais fácil inferir o quão semelhante eles são. A semelhança entre eles é dada pelo fato de representarem uma mesma entidade no mundo real. As três técnicas deste tipo diferem na forma que usam as instâncias para comparar os conceitos.

- Comparação das instâncias comuns: compara o número de instâncias comuns entre os dois conceitos e usa isso para inferir a semelhança.
- Identificação de instâncias: essa técnica é usada principalmente para encontrar semelhanças entre propriedades. Uma instância em comum entre os dois conceitos é identificada e, analisando os valores de cada propriedade, são deduzidas quais propriedades são semelhantes.
- Disjunção: ao mesmo tempo que se pode inferir semelhança entre conceitos analisando as instâncias em comum, pode-se seguir o caminho oposto, analisando o quão disjuntos são dois conjuntos de instâncias para calcular que não são semelhantes.

2.3.4 Técnicas Semânticas

Técnicas baseadas em semântica caracterizam-se por fazerem uso de um modelo teórico para justificar as correspondências semânticas encontradas. Essas técnicas geralmente requerem *seed alignments* (por exemplo, entre as ontologias sendo comparadas e uma ontologia de domínio). *Seed alignments* são alinhamentos que podem conter conhecimento sobre o domínio que está se trabalhando, sendo usado para a inferência de

conhecimento adicional. Ainda usando a classificação descrita em [Euzenat and Shvaiko 2007a], as técnicas semânticas se dividem em dois tipos:

Técnicas Dedutivas

Requer um pré-processamento no qual é feito um *merging* entre as duas ontologias comparadas. Os novos relacionamentos encontrados no processo de *merging* das duas ontologias são usados, aplicando-se regras da lógica descritiva para inferir mais relacionamentos.

Background Knowledge

Técnicas que fazem uso de *background knowledge* tentam suprir a falta de conhecimento interno às duas ontologias comparadas, explorando um recurso externo para diminuir a lacuna semântica entre elas [Sabou *et al.* 2008]. Ao contrário do método semântico, não são fornecidos alinhamentos de entrada e a ontologia externa não é passada como parâmetro. O *matcher* semântico pode buscar em repositórios de ontologias na Internet como o *Swoogle* [Swoogle 2011] ou o *Watson* [d'Aquin *et al.* 2007].

Existem trabalhos [Sabou *et al.* 2008] que estudam formas de buscar não apenas uma ontologia externa, mas várias, e de encontrar relacionamentos usando-as como *background knowledge*. Esses trabalhos alegam que, quando se depende de apenas uma fonte externa, geralmente tenta-se usar alguma ontologia muito genérica, e perde-se o conhecimento específico do domínio onde se está trabalhando.

Pode-se inferir relacionamentos dentro de apenas uma ontologia, ou encontrar relacionamentos que estão em diferentes ontologias. Em ambos os casos são consultadas várias ontologias. Na Figura 2.3, várias ontologias externas são consultadas. A partir dos *seed alignments* onde estariam inclusas as correspondências $A \rightarrow A_2'$, $A \rightarrow A_3'$, $B \rightarrow B_2'$, $B \rightarrow B_3'$, pode-se inferir que aquela correspondência é verdadeira entre aqueles elementos.

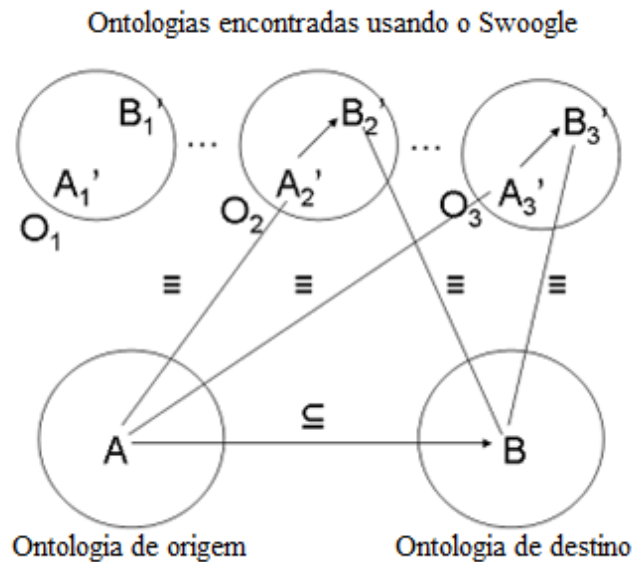


Figura 2.3 – Inferência de correspondência consultando apenas uma ontologia [Sabou *et al.* 2008].

Na Figura 2.4, o modo de inferir correspondências é bem semelhante, porém ao invés de procurar por relacionamentos entre os conceitos em apenas uma ontologia externa por vez, é feita a busca em várias por vez, buscando relacionamentos que só podem ser descobertos se consultadas mais de uma ontologia externa.

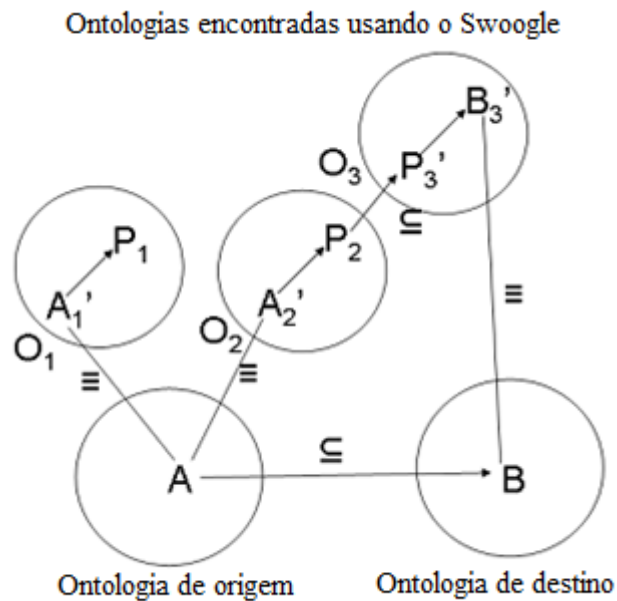


Figura 2.4 – Inferência de correspondências consultando mais de uma ontologia [Sabou *et al.* 2008].

Contexto de um Conceito

As técnicas que não levam em consideração o contexto do conceito geralmente usam comparação de *string* sem analisar a estrutura das ontologias como um todo (no caso

das técnicas estruturais, só se analisa a distância entre os conceitos no grafo da ontologia). Muitas vezes os superconceitos nos dão informações importantes sobre o significado do rótulo que representa o conceito que está sendo analisado.

A análise dos superconceitos de um conceito analisado é chamada de contexto do conceito. Existem vários trabalhos que usam esta ideia como o S-Match [Giunchiglia *et al.* 2009], C-SAW [Tierney and Jackson 2008], ctxMatcher [Bouquet *et al.* 2003]. A Figura 2.5 mostra um caso simples de desambiguação feita analisando o contexto do conceito. O conceito *Macaco* no exemplo (a) refere-se ao animal macaco, que é um mamífero. No exemplo (b), *Macaco* refere-se à ferramenta macaco, usada em carros para auxiliar a troca de pneus. Essa conclusão é tirada observando-se que o seu superconceito é *Ferramentas*.

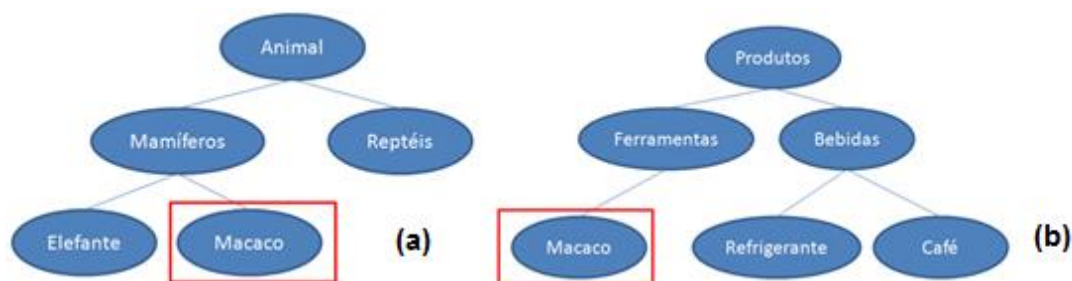


Figura 2.5 – Exemplo de desambiguação feita usando análise do contexto do conceito.

Histórico e Feedback do Usuário

Informações sobre *matchings* passados (por exemplo, alinhamentos anteriores) podem ser armazenadas em um histórico para melhorar a qualidade de futuras operações de *matching*. Os alinhamentos armazenados podem ser usados para corrigir correspondências que em um processo atual sejam muito diferentes das encontradas anteriormente para certo par de conceitos.

Também pode ser oferecida ao usuário a possibilidade de avaliar correspondências de um alinhamento. A ferramenta de *matcher* armazena esse *feedback* e, com base nesse novo conhecimento, quando gerar novas correspondências, poderá rejeitar ou adicionar algumas correspondências dependendo da avaliação feita pelo usuário.

Alguns trabalhos [Paulheim *et al.* 2007, Rahm and Bernstein 2001] analisam como esse conhecimento passado pelo usuário pode ser usado. Em geral, o valor de similaridade entre os conceitos e este pode ser diminuído se muitos usuários avaliarem negativamente uma correspondência no passado. Assim, correspondências irrelevantes seriam descartadas após certo número de avaliações negativas. Trabalhos recentes

ênfatizam o uso de *feedback* do usuário no processo de *matching* corrente [Belhajjame *et al.* 2010] [Duan *et al.* 2010] para tentar descartar ou inferir outros relacionamentos de forma iterativa, onde o usuário interage informando correspondências incorretas e a ferramenta tenta inferir que outras correspondências estão incorretas a partir desta informada. Novas correspondências são retornadas e o usuário pode avaliar novamente. O processo se repete até o usuário aceitar o alinhamento.

2.4 Usos de *Matching* de Ontologias

O *matching* de ontologias é uma operação vital para alguns tipos de sistema. Um exemplo deles são os sistemas de integração de informações [Halevy *et al.* 2005]. Inicialmente a operação de *matching* era usada em cenários menores, como a integração de esquemas em um caso onde se necessita integrar dados contidos em dois bancos de dados que possuem informações de domínios semelhantes, mas apresentam esquemas diferentes. A identificação das semelhanças entre entidades e atributos pode ser feita com o auxílio de um processo de *ontology matching*, auxiliando o trabalho de integração de informações.

Em um passo seguinte, a operação de *matching* de ontologias começou a ser usada para fazer a integração dos próprios dados [Halevy *et al.* 2005]. Nesse cenário, são usadas ontologias locais para representar cada fonte de dados, e o acesso a informações é feito com consultas submetidas a uma ontologia virtual, que é uma ontologia que funciona como uma visão global sobre todas as bases integradas, geralmente obtida através de uma união das ontologias que representam essas bases. Quando uma consulta é executada sobre a ontologia virtual, operações de *matching* são realizadas entre a ontologia virtual e as ontologias locais para se identificar correspondências e assim poder reescrever as consultas de acordo com o esquema local de cada base.

Com o aumento do número de bases de dados e o crescimento das redes *Peer-to-Peer* (P2P), um novo cenário surgiu para o compartilhamento de dados. A integração e o compartilhamento de informações em uma rede P2P é uma tarefa ainda mais complexa do que nos casos anteriores. O ambiente P2P traz as dificuldades não só pelo aumento do número de fontes de dados, como também pelos problemas inerentes a esse tipo de rede como o fato de nem sempre todas as fontes estarem disponíveis e a alta dinamicidade das fontes entrando e saindo da rede.

Ao mesmo tempo, Sistemas P2P de Integração de Dados (*Peer Data Management Systems* - PDMSs) [Sung *et al.* 2006] foram desenvolvidos com um

conceito parecido, fazendo uso de esquemas locais, onde podem ser feitas consultas locais, que são executadas apenas no próprio *peer*, ou consultas globais. Para as consultas globais, elas são reformuladas em consultas locais para outros *peers*. Em PDMSs que usam ontologias [Xiao 2006], ferramentas de *ontology matching* são usadas para inferir relacionamentos entre as entidades dos dois esquemas e ajudar na reformulação.

2.5 Métodos para Avaliação de Alinhamentos

Até 2002, poucos eram os métodos para avaliar a qualidade dos alinhamentos das ferramentas desenvolvidas até então. Neste ano, aconteceu pela primeira vez o *Evaluation of Ontology-based Tools* (EON), com o objetivo de criar experimentos para a avaliação de alinhamentos gerados por ferramentas de *ontology matching*.

Cada um desses experimentos, chamados *benchmarks*, tentava avaliar uma característica específica de cada *ontology matching*, identificando assim seus pontos fortes e fracos. Em paralelo a esse evento, outro evento chamado I3CON era realizado todo ano, com o mesmo objetivo. Em 2005, esses dois eventos foram unificados em apenas um, o *Workshop on Integrating Ontologies*, organizado pela OAEI (*Ontology Alignment Evaluation Initiative*) [Euzenat et al. 2005] [OAEI 2011].

As avaliações feitas das ferramentas usavam medidas que analisavam o grau de conformidade dos alinhamentos com o que era esperado, medidas de desempenho, dentre outros. Nas próximas seções serão detalhadas as medidas de avaliação mais importantes usadas pela OAEI [Euzenat et al. 2005; OAEI].

2.5.1 Medidas de Conformidade

Essas medidas consistem em comparar o alinhamento A gerado por uma ferramenta com um alinhamento de referência R , também chamado de *gold alignment*. Dentre elas as mais usadas são a precisão e a cobertura.

- **Precisão**

A precisão representa a fração de correspondências retornadas que é relevante. Em outras palavras, dentre as correspondências retornadas, quantas são corretas em relação ao alinhamento de referência. O valor retornado está entre 0 (zero) e 1 (um), sendo 0 o pior caso, e 1 o melhor. O resultado é calculado dividindo-se a quantidade de correspondências na interseção entre as correspondências de referência (R) com as correspondências geradas (A), dividido pela quantidade de correspondências geradas, como vemos na Equação 2.1.

$$\text{Precisão} = \frac{|R \cap A|}{|A|} \quad (2.1)$$

- **Cobertura**

A cobertura representa a fração de documentos que são relevantes que são trazidos. Em outras palavras, das correspondências consideradas corretas pelo alinhamento de referência, quantos foram retornados pela ferramenta. O valor retornado está entre 0 (zero) e 1 (um), sendo 0 o pior caso, e 1 o melhor. O resultado é calculado dividindo-se a quantidade de correspondências na interseção entre as correspondências de referência (R) com as correspondências geradas (A), dividido pela quantidade de correspondências de referência, como vemos na Equação 2.2.

$$\text{Cobertura} = \frac{|R \cap A|}{|R|} \quad (2.2)$$

- **Fallout**

O *fallout* é uma medida de porcentagem das correspondências que são falsos positivos. Em outras palavras, quantas das correspondências encontradas (A) não estão no alinhamento de referência (R). O valor retornado está entre 0 (zero) e 1 (um), sendo 0 o melhor caso, e 1 o pior caso. O resultado é calculado subtraindo das correspondências geradas, as que fazem parte do alinhamento de referência, e dividindo essa quantidade pelo número de correspondências geradas, como vemos na Equação 2.3.

$$\text{Fallout} = \frac{|A| - |A \cap R|}{|A|} = \frac{|A / R|}{|A|} \quad (2.3)$$

- **F-measure**

Como uma forma de ter uma medida única considerando tanto precisão, como cobertura, foi desenvolvida a medida *f-measure*. A *f-measure* é uma média harmônica entre precisão e cobertura. O α na fórmula é um valor entre 0 (zero) e 1 (um), e atribui um peso maior a precisão ou a cobertura. Normalmente o valor utilizado é 0,5 [Euzenat and Shvaiko 2007a]. O *f-measure* é calculado usando a fórmula mostrada na Equação 2.4.

$$\text{F-measure} = \frac{\text{Precisão} \times \text{Cobertura}}{(1 - \alpha) \times \text{Precisão} + \alpha \times \text{Cobertura}} \quad (2.4)$$

2.5.2 Medidas de Desempenho

Ao contrário das medidas de conformidade, as medidas de desempenho dependem de um cenário que envolve o *hardware* usado, a qualidade da banda disponível para acesso a Internet, dentre outros fatores. A seguir discutimos algumas dessas medidas.

- **Tempo de Processamento**

A velocidade é medida como o tempo de execução de todo o processo de *matching*. Se uma interação com o usuário é necessária, o tempo durante a interação não é contado. O tempo é medido em segundos ou em minutos no caso das ferramentas que demoram mais a executar.

- **Consumo de Memória**

A quantidade de memória principal (*Random Access Memory* - RAM) usada para executar o processo de *matching* é outra medida possível de ser mensurada, em *Megabytes*. Se há uma dependência de alguma outra ferramenta, apenas o valor adicional deve ser utilizado.

2.6 Considerações Finais

Neste capítulo, apresentamos uma visão geral sobre os conceitos e estudos sobre *ontology matching* que são a base para o entendimento deste trabalho. No próximo capítulo, veremos ferramentas de *ontology matching*, comparando as técnicas usadas por cada uma e identificando aspectos a serem melhorados.

3. Ferramentas de *Ontology Matching* Existentes

Várias são as ferramentas de *ontology matching* existentes, cada uma delas faz uso de uma ou mais das técnicas apresentadas no capítulo anterior. Nas próximas seções serão analisadas as principais ferramentas. Para a escolha dos *matchers* a serem analisados, selecionamos algumas ferramentas já consolidadas como o CUPID [Madhavan *et al.* 2001] e o COMA++ [Aumüller *et al.* 2005], mais outras ferramentas que usam algumas das novas funcionalidades que serão adicionadas ao *SemMatcher+* como: C-SAW [Tierney and Jackson 2008], S-Match [Giunchiglia *et al.* 2009], ctxMatcher [Bouquet *et al.* 2003] e SCARLET [Sabou *et al.* 2007]. Além desses escolhemos dois *matchers* do último *benchmark* da OAEI (2010), o *GeRoMeSuite* [Quix *et al.* 2010] e o ASMOV [Jean-Mary *et al.* 2010]. Vamos falar sobre cada um deles nas seções a seguir.

3.1 CUPID

O CUPID [Madhavan *et al.* 2001] foi desenvolvido em conjunto pela Universidade de Washington, Microsoft e Universidade de Leipzig e combina técnicas linguísticas e estruturais de *matching*. O processo de *matching* consiste de duas fases como vemos na Figura 3.1.

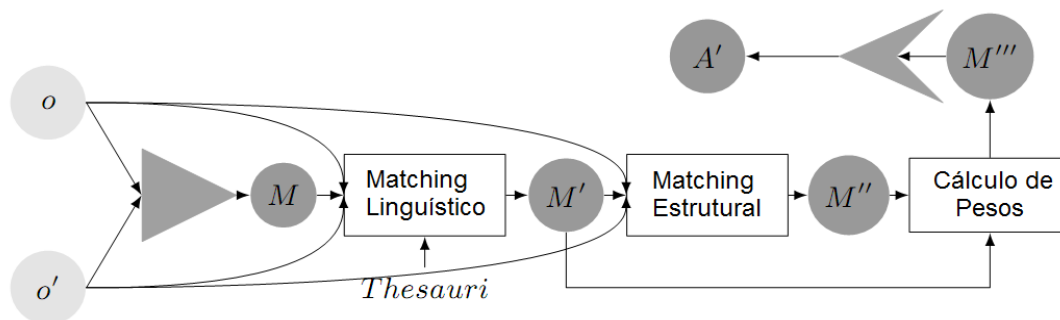


Figura 3.1 – Processo de *matching* do CUPID [Euzenat and Shvaiko 2007a]

Em uma primeira etapa, técnicas linguísticas são usadas para descobrir correspondências baseadas no nome (dos conceitos ou propriedades), no seu tipo, domínio, entre outros. Os conceitos e propriedades passam por um processo de normalização, onde são removidos artigos e preposições, abreviações são expandidas e palavras compostas são tratadas separadamente. *Thesauri* são usados para detectar o uso de sinônimos e abreviações. Apenas depois dessa normalização os conceitos e propriedades são comparados e o alinhamento linguístico é gerado.

Na segunda fase, técnicas estruturais são usadas para descobrir mais alinhamentos. Para cada conceito, são analisados os superconceitos em comum. Observamos o uso de contexto do conceito como na técnica descrita na Seção 2.3.4.3, apesar de ainda não usar essa denominação. No final dessa fase temos como resultado o alinhamento estrutural.

Na fase final, os alinhamentos gerados pelas duas etapas anteriores (linguístico e estrutural) são combinados calculando-se uma média ponderada entre cada alinhamento. Os pesos da média ponderada são configuráveis.

3.2 COMA++

A maioria das ferramentas de *matcher* tendem a implementar apenas um conjunto pequeno de técnicas. O COMA++ [Aumuellner *et al.* 2005] segue um caminho um pouco diferente, e oferece a possibilidade de usar várias técnicas já desenvolvidas de *ontology matching*, e fazer combinação de seus resultados.

A escolha e parametrização das técnicas usadas são feitas por meio de uma interface gráfica que é a melhor entre as ferramentas existentes. Também há a possibilidade de reuso de alinhamentos. Na Figura 3.2 vemos a arquitetura do COMA++.

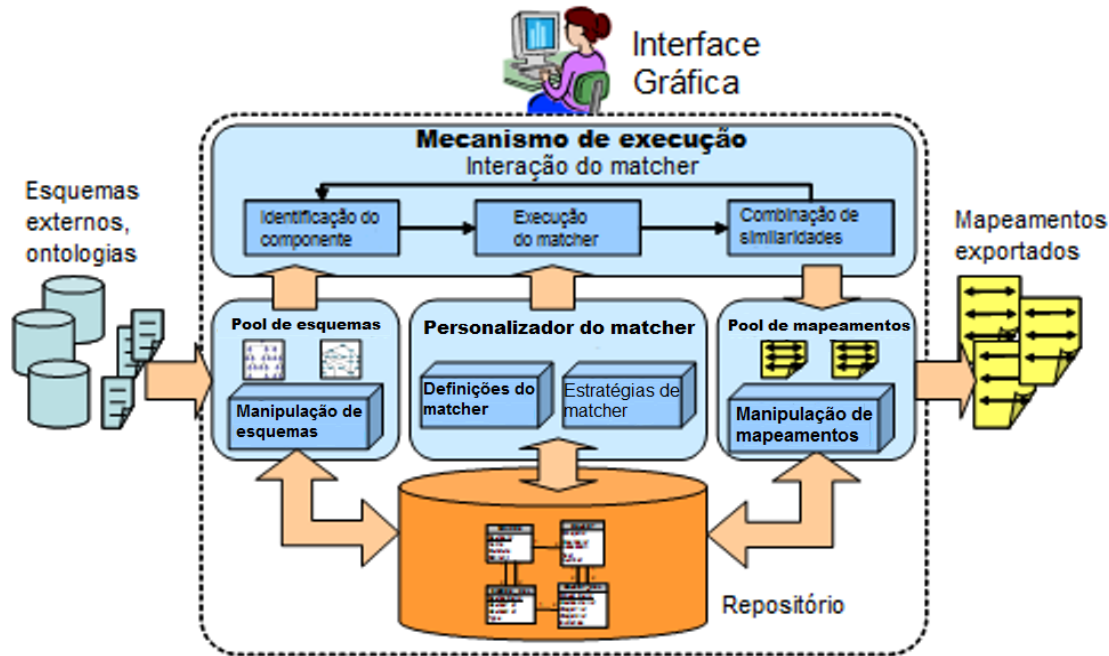


Figura 3.2 – Arquitetura do COMA++ [Aumuellner *et al.* 2005]

O repositório armazena dados relacionados aos *matchings* já executados. O *pool de esquemas* e o *Pool de mapeamentos* gerenciam esquemas, ontologias e mapeamentos na memória principal. O *Personalizador do matcher* é o componente que permite a

escolha dos *matchers* e estratégias de *matching* a serem utilizadas. O histórico criado pelo COMA++ armazena os alinhamentos no repositório e os usa posteriormente sem avaliar o domínio das ontologias envolvidas.

3.3 C-SAW

O C-SAW [Tierney and Jackson 2008] é uma ferramenta que gera o que se chama de *contextual semantic alignment*, considerando o contexto de cada conceito nas ontologias que estão sendo comparadas. Para descobrir o contexto do conceito, a técnica de *background knowledge* é usada. Nesse caso, o *WordNet* é usado para encontrar o *Semantic Alignment Words* (SAW) de cada conceito das ontologias.

Para encontrar a semelhança entre dois conceitos, é encontrado o SAW de cada um deles, que consiste no conjunto de conceitos desde o próprio conceito até a raiz (o conjunto de superconceitos), representando o contexto do conceito. Compara-se o SAW de cada conceito com SAW do mesmo conceito no *WordNet*, encontrando-se o caminho de superconceitos no *WordNet* que corresponde ao real sentido do conceito. Isto evita que conceitos com grafias semelhantes sejam ditos equivalentes, mas estejam se referindo a conceitos diferentes no mundo real.

Também é feito o chamado reforço semântico negativo, que busca não apenas saber que palavras são próximas ao conceito comparado (o SAW), mas também analisar as palavras que não são próximas (ou seja, os caminhos de conceitos no *WordNet* que tratam da mesma palavra mas em outro contexto) como uma forma de reforçar o significado daquele conceito.

3.4 S-Match

A ferramenta S-Match [Giunchiglia *et al.* 2009] enfatiza a questão de não comparar apenas o *label* das classes ou propriedades da ontologia, mas analisar o conceito que ele representa no mundo real. A ferramenta aponta como um erro dos *matchers* tradicionais o fato de compararem apenas os *labels* e não o que eles realmente significam analisando os superconceitos em comum nas duas ontologias. Também é possível consultar uma fonte externa que, como o C-SAW, também é feito analisando os superconceitos em comum no *WordNet*.

Há uma etapa de pré-processamento onde, para cada classe ou propriedade, é extraído apenas o radical de seus *labels* e, em seguida, para cada par comparado é extraído seu contexto consultando os superconceitos em comum (que também são comparados na forma de radical). De acordo com a quantidade de superconceitos em

comum, a equivalência entre os termos é calculada. Um dos futuros trabalhos relacionados à ferramenta é o uso do *feedback* do usuário para melhoria das correspondências geradas.

3.5 ctxMatcher

A ferramenta ctxMatcher [Bouquet *et al.* 2003] explora técnicas linguísticas, de domínio e estruturais na descoberta de correspondências. O algoritmo possui dois passos, a explicitação semântica e a comparação semântica.

No primeiro passo, usa-se um conhecimento externo (*WordNet* ou ontologia de domínio, no caso do ctxMatcher, *WordNet*) para buscar os possíveis significados para o rótulo de cada conceito das duas ontologias de entrada, sendo chamado esse processo de interpretação linguística.

Ainda no primeiro passo é feita a chamada *contextualização*. A partir da interpretação linguística, são filtrados os significados que realmente definem cada conceito. Essa filtragem é feita comparando a semelhança entre os significados e o foco do conceito em sua ontologia. O foco corresponde ao conjunto de conceitos da raiz até o significado, e também os seus conceitos filhos. A ideia de analisar a estrutura é inspirada em como os próprios humanos fazem análise da estrutura para comparar os conceitos [Bouquet *et al.* 2003].

A segunda fase, chamada comparação semântica, usa conhecimento externo (mais uma vez o *WordNet*) para inferir relacionamentos semânticos usando regras. As regras são aplicadas sobre significados filtrados na fase anterior e inferem relacionamentos de equivalência, subconceito, superconceito e disjunção.

3.6 SCARLET Matcher

O SCARLET (*SemantiC relAtion discoveRy by harvesting onLinE onTologies*) Matcher [Sabou *et al.* 2007], faz parte de uma nova geração de ferramentas de *matching* que exploram a crescente quantidade de ontologias disponíveis na Internet. O SCARLET é parte do Projeto NeOn [NeOn 2011], que tem como objetivo dar suporte a tarefas envolvendo ontologias.

A Web Semântica é usada como *background knowledge* para inferir relacionamentos semânticos, por meio de sistemas de busca como o *Watson* [d'Aquin *et al.* 2007] e *Swoogle* [Swoogle 2011]. Por meio desses engenhos de busca, são encontradas ontologias com conceitos de nome igual aos que estão sendo comparados, e o tipo de correspondência entre eles é descoberto verificando o tipo de relacionamento

nas ontologias encontradas. O processo de descoberta de alinhamentos feito pelo SCARLET foi explicado na Seção 2.3.4.

3.7 GeRoMeSuite

O *GeRoMeSuite* [Quix *et al.* 2010] foi uma das ferramentas apresentadas na edição 2010 do OAEI, e consiste de um sistema genérico que efetua operações de *matching* não apenas entre ontologias, mas também entre esquemas XML e até mesmo SQL.

O processo de *matching* ocorre em duas fases. A primeira, usando uma ontologia obtida na Internet, por meio do *Swoogle* e do *Google*. Consultas são geradas usando-se o rótulo dos conceitos das duas ontologias de entrada. A partir do resultado, uma ontologia é escolhida para servir de *background knowledge* e relacionamentos são inferidos como acontece no SCARLET.

A segunda fase usa aprendizagem de máquina para identificar relacionamentos semânticos. A verificação de relacionamentos de subconceito é tratada como um problema de classificação binária, ou seja, um par de conceitos pode ser classificado como subconceito ou não. A estrutura das ontologias de entrada é analisada e, a partir de suas hierarquias, os relacionamentos de subconceito são usados como conjunto de treinamento, e os relacionamentos encontrados na primeira fase são verificados de acordo com o classificador, obtendo-se o alinhamento final.

3.8 ASMOV

O ASMOV [Jean-Mary *et al.* 2010] é uma ferramenta de *ontology matching* que foi desenvolvida para facilitar a integração de fontes de dados modeladas como ontologias, e também foi uma das ferramentas apresentadas no OAEI 2010.

Sua abordagem usa uma média ponderada das medidas de similaridade de quatro diferentes características das ontologias: elementos léxicos (identificador, rótulo e comentários), estrutura relacional (hierarquia), estrutura interna (propriedades, tipo das propriedades e restrições) e extensão (instâncias das classes). No caso da semelhança léxica, o *WordNet* é utilizado para verificar a semelhança entre os conceitos. A partir do cálculo das características, obtém-se um pré-alinhamento.

O pré-alinhamento é então enviado para a fase de verificação semântica, onde inconsistências são identificadas e removidas. São cinco os tipos de inconsistências detectadas pelo ASMOV:

1. Múltiplas correspondências com a mesma entidade: quando uma entidade tem vários relacionamentos com entidades diferentes de outra ontologia;

2. Hierarquia inconsistente: quando um conceito c_1 na ontologia O_1 é equivalente a um conceito c_1' na ontologia O_2 , há uma incoerência se um subconceito de c_1 é equivalente a um superconceito de c_1' ;
3. Disjunção inconsistente: quando dois conceitos c_1 e c_2 são disjuntos na ontologia O_1 , não pode haver um relacionamento de equivalência entre eles e dois conceitos c_1' e c_2' da ontologia O_2 respectivamente, se eles são subconceito um do outro;
4. *Subsumption* inconsistente: quando dois conceitos c_1 e c_2 na ontologia O_1 são equivalentes a dois conceitos c_1' e c_2' da ontologia O_2 respectivamente, se c_1 é superconceito de c_2 , c_2' não pode ser superconceito de c_1' ;
5. Propriedades inconsistentes: se duas propriedades são equivalentes, os conceitos aos quais elas estão associadas também devem ser equivalentes.

Removidas as inconsistências, as correspondências restantes formam o alinhamento final. Esta ferramenta obteve bons resultados no OAEI. Os pesos da média ponderada da fase inicial foram escolhidos testando os alinhamentos gerados com o alinhamento de referência da edição anterior do OAEI.

3.9 SemMatcher

O *SemMatcher* em sua primeira versão [Pires *et al.* 2009; Pereira 2008] combina estratégias de *matching* já conhecidas. Um *matcher* semântico e outro *matcher* linguístico estrutural são executados em paralelo. Os valores de similaridade obtidos de ambos os *matchers* são combinados por uma média ponderada. O peso de cada *matcher* pode ser configurado. Na Figura 3.3 vemos a arquitetura do processo.

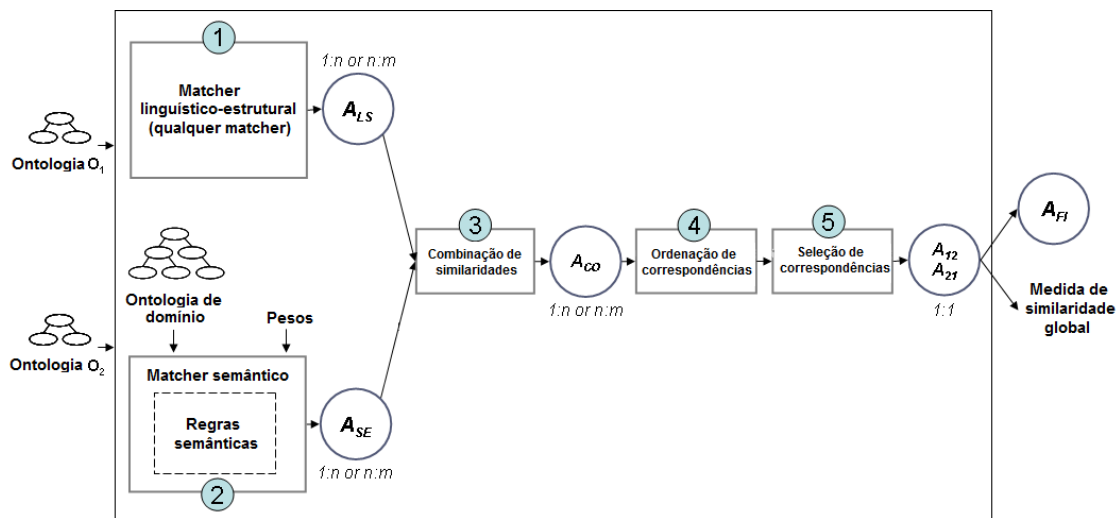


Figura 3.3 – Arquitetura da primeira versão do *SemMatcher* [Pires *et al.* 2009]

Duas ontologias são passadas como entrada, sendo as ontologias a serem comparadas (*Local Ontologies* ou LOs, na Figura 3.3 elas são O_1 e O_2). Além dessas, uma ontologia de domínio (*Domain Ontology* ou DO) é usada como *background knowledge* no *matcher* semântico. A seguir cada etapa mostrada na Figura 3.3 é explicada:

(1) Nesta fase, qualquer *matcher* linguístico-estrutural existente pode ser usado para gerar o alinhamento A_{LS} . O *SemMatcher* permite o uso do AlignmentAPI [Euzenat 2004] ou do H-match [Castano *et al.* 2003] pela própria ferramenta. Além disso, pode ser usada qualquer ferramenta que gere alinhamentos em um arquivo no formato definido pela OAEI [OAEI 2011]. O resultado dessa etapa é o alinhamento A_{LS} .

(2) Usando uma ontologia do domínio associado, o *matcher* semântico identifica as equivalências entre as ontologias a serem comparadas (LOs) e a DO. A partir dessas equivalências, são aplicadas regras para encontrar relacionamentos semânticos entre os conceitos das ontologias (LOs). No total, sete regras foram implementadas, sendo que três delas funcionam igualmente para conceitos e propriedades [Souza 2009]. O Resultado da execução do *matcher* semântico é o alinhamento A_{SE} .

(3) Nesta etapa, os valores individuais das correspondências geradas nas fases anteriores (A_{LS} e A_{SE}) são combinados usando uma média ponderada de seus respectivos valores de similaridade, tendo como saída o alinhamento combinado A_{CO} .

(4) Correspondências contendo elementos de O_1 são ranqueadas em ordem decrescente de acordo com os valores de similaridade associados.

(5) Uma seleção das correspondências é feita, filtrando para cada conceito de O_1 , a melhor correspondência (aquela que possui o maior valor de similaridade). O resultado da aplicação deste filtro é a geração do alinhamento A_{12} .

Os passos 4 e 5 são executados também na direção oposta (em relação a O_2), gerando no final o alinhamento A_{21} , neste caso, também é usado o alinhamento A_{CO} , porém com os relacionamentos invertidos. Consideramos nesta inversão que a equivalência continua, pois é um relacionamento igual nos dois sentidos, já um relacionamento de subconceito, se torna superconceito. No caso da composição, "parte de" se torna "todo de" e o *Closeness* e a disjunção continuam com o mesmo significado. Como saída, são gerados um alinhamento final (união dos alinhamentos A_{12} e A_{21}), e uma medida de similaridade entre as ontologias calculada usando os alinhamentos A_{12} e A_{21} . Para o cálculo, são usadas as medidas de similaridade existentes *dice* [Aumuellner *et al.* 2005], *weighted* [Castano *et al.* 1998] e *overlap* [Rijsbergen 1979]. O tamanho da

ontologia é determinado pelo seu número de conceitos, representado nas fórmulas por $|O|$. Na medida *dice*, a quantidade de correspondências encontradas em ambas as direções são somadas e o divididas pelo total de conceitos em cada ontologia. Já a medida *weighted*, ao invés de simplesmente usar a quantidade de correspondências, faz um somatório de cada grau de similaridade em ambas as direções. O resultado é também dividido pela quantidade de conceitos em ambas as ontologias. Para calcular a medida *overlap*, dividimos a quantidade de correspondências presentes em ambas as direções pelo tamanho da menor ontologia. As fórmulas para cada medida são mostradas na Figura 3.11.

$$Dice(O_1, O_2) = \frac{|A_{12}| + |A_{21}|}{|O_1| + |O_2|} \quad Weighted(O_1, O_2) = \frac{\sum_{j=1}^{|A_{12}|} n + \sum_{j=1}^{|A_{21}|} n}{|O_1| + |O_2|} \quad Overlap(O_1, O_2) = \frac{|A_{12} \cap A_{21}|}{\min(|O_1|, |O_2|)}$$

Figura 3.11 – Medidas disponíveis no SemMatcher [Pires et al. 2009]

Como explicado anteriormente, sete regras foram implementadas no *matcher* semântico. Foram elas:

- **Equivalência**

Na abordagem usada, dois elementos são considerados equivalentes se eles representam um mesmo conceito ou propriedade no mundo real.

Definição 1 - Regra *isEquivalentTo*: um elemento $O_1:x$ é equivalente a $O_2:y$ se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:k$, onde x , y e k são todos conceitos ou propriedades. Esta correspondência é representada por $O_1:x \equiv O_2:y$ como mostrado na Figura 3.4.

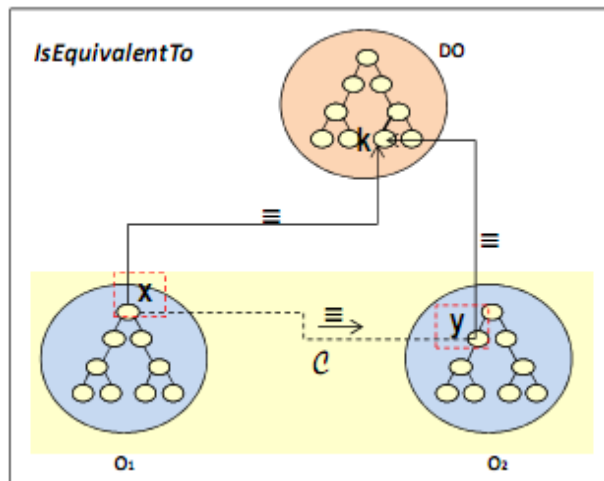


Figura 3.4 – Correspondência do tipo *isEquivalentTo* [Souza 2009]

- **Especialização e generalização**

Para representar a correspondência de *isSubConceptOf* usamos $O_1:x \sqsubseteq O_2:y$, neste caso, o elemento x é menos genérico que o conceito y . Já o *isSuperConceptOf* é

representado por $O_1:x \supseteq O_2:y$, indicando que o elemento x é mais genérico que o elemento y .

Definição 2 - Regra *isSubConceptOf*: um elemento $O_1:x$ é subconceito de $O_2:y$ se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:z$ e $DO:k \subseteq DO:z$, onde x, y, k e z são todos conceitos ou propriedades. Esta correspondência é representada por $O_1:x \sqsubseteq O_2:y$ como mostrado na Figura 3.5.

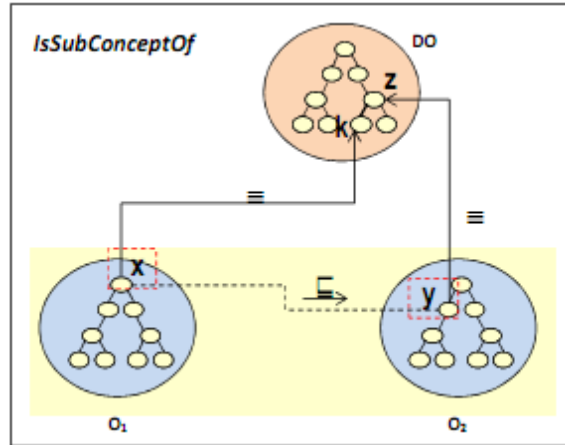


Figura 3.5 – Correspondência do tipo *isSubConceptOf* [Souza 2009]

Definição 3 - Regra *isSuperConceptOf*: um elemento $O_1:x$ é subconceito de $O_2:y$ se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:z$ e $DO:k \supseteq DO:z$, onde x, y, k e z são todos conceitos ou propriedades. Esta correspondência é representada por $O_1:x \supseteq O_2:y$ como mostrado na Figura 3.6.

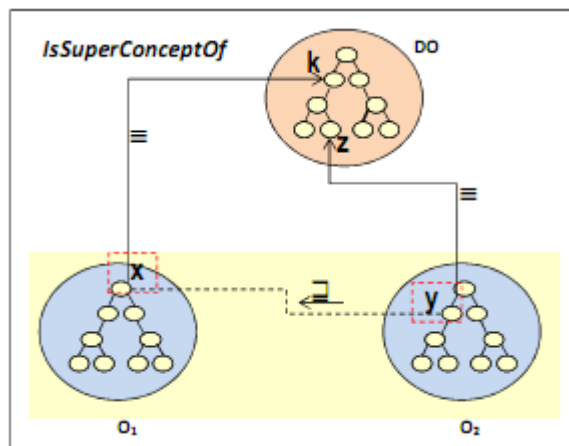


Figura 3.6 – Correspondência do tipo *isSuperConceptOf* [Souza 2009]

- **Agregação**

Para os relacionamentos de agregação, apenas conceitos são considerados. Dois tipos de correspondências foram definidos. Se um conceito x em O_1 é idêntico ao conceito k na DO , e o conceito y em O_2 é idêntico a um conceito z na DO , e, na DO , k é parte de z , o conceito x é parte de y . Se um conceito x em O_1 é idêntico ao conceito k na

DO, e o conceito y em O_2 é idêntico ao conceito z na DO, e, na DO, k é todo de z , o conceito x é conjunto de y .

Definição 4 - Regra *isPartOf*: um conceito $O_1:x$ é parte de $O_2:y$, se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:z$ e $DO:k \triangleright DO:z$. Este relacionamento é representado por $O_1:x \triangleright O_2:y$ como mostrado na Figura 3.7.

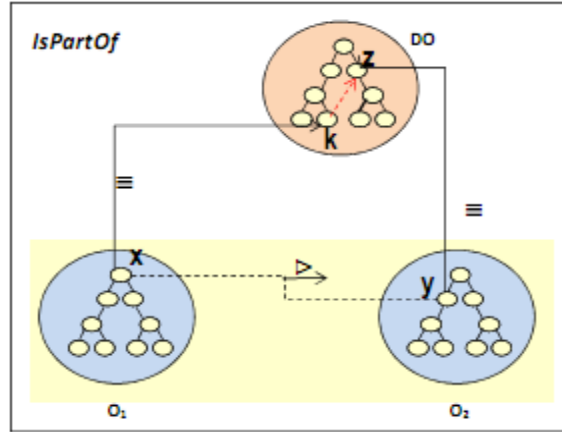


Figura 3.7 – Correspondência do tipo *isPartOf* [Souza 2009]

Definição 5 - Regra *isWholeOf*: um conceito $O_1:x$ é todo de $O_2:y$, se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:z$ e $DO:k \triangleleft DO:z$. Esta correspondência é representada por $O_1:x \triangleleft O_2:y$ como mostrado na Figura 3.8.

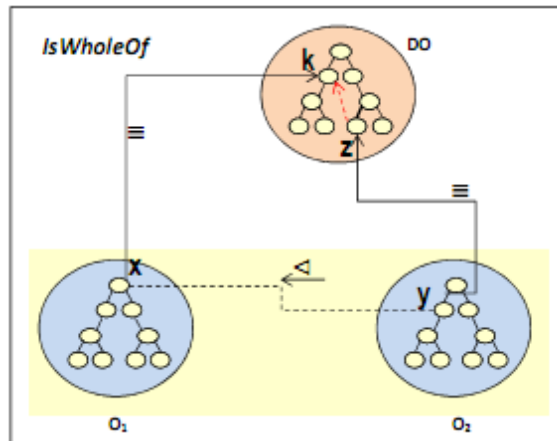


Figura 3.8 – Correspondência do tipo *isWholeOf* [Souza 2009]

- **Proximidade**

São considerados próximos, dois conceitos que possuam um contexto ou significado comum no mundo real. Se há um conceito x em O_1 que é idêntico a um conceito k na DO, e outro conceito y em O_2 é idêntico a um conceito z na DO. E na DO, k e z são subclasses de um conceito a em comum, esse conceito a não é o *root*, e k e z

estão até no máximo n níveis de profundidade abaixo de a (onde n é o valor do limiar), o conceito x é próximo de y .

Definição 6 - Regra *isCloseTo*: um conceito $O_1:x$ é próximo de $O_2:y$, se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:z$ e $DO:k \subseteq DO:a$ e $DO:z \subseteq DO:a$ e $DO:a \neq \text{raiz}$ e $\text{profundidade}(DO:k, DO:a) < \text{limiar}$ e $\text{profundidade}(DO:z, DO:a) < \text{limiar}$. Como mostrado na Figura 3.9.

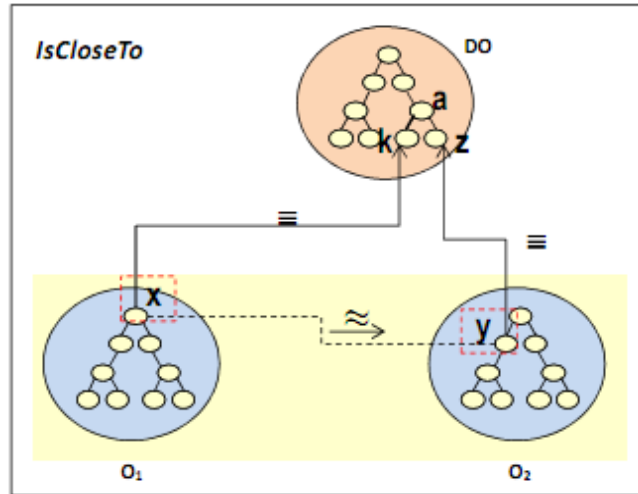


Figura 3.9 – Correspondência do tipo *isCloseTo* [Souza 2009]

- **Disjunção**

Dois conceitos são considerados disjuntos na abordagem do *SemMatcher*, quando eles não apresentam nenhuma semelhança em seu significado. Se há um conceito x em O_1 que é idêntico a um conceito k na DO , e outro conceito y em O_2 é idêntico a um conceito z na DO . E na DO , k e z são disjuntos, x é disjunto y .

Definição 7 - Regra *isDisjointWith*: um conceito $O_1:x$ é disjunto de $O_2:y$, se $O_1:x \equiv DO:k$ e $O_2:y \equiv DO:z$ e $DO:k \perp DO:z$. Esta correspondência é representada por $O_1:x \perp O_2:y$. Como mostrado na Figura 3.10.

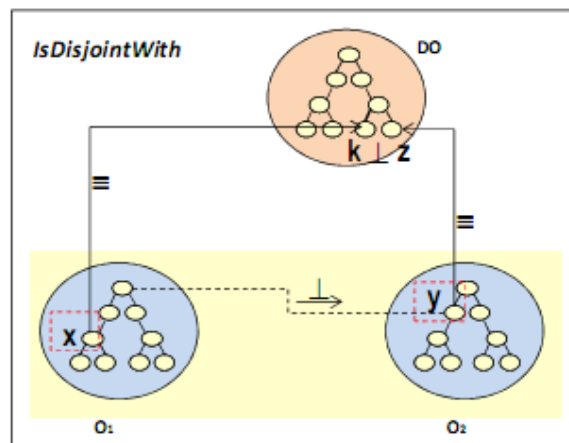


Figura 3.10 – Correspondência do tipo *isDisjointWith* [Souza 2009]

3.10 Considerações Finais

Neste capítulo, vimos algumas das ferramentas de *ontology matching* mais utilizadas. Também foi detalhada a ferramenta *SemMatcher* em sua primeira versão, antes das mudanças propostas neste trabalho. Observando as técnicas usadas nos *matchers* mostrados, vimos que algumas delas podem ser incorporadas ao *SemMatcher* para melhorar a qualidade dos alinhamentos. Fazemos a seguir um quadro comparativo com essas funcionalidades.

Ferramenta	<i>WordNet</i>	Ontologia externa	Feedback do usuário
CUPID	Para extrair radicais dos rótulos.	Não	Não
COMA++	Não	Não	Não
C-SAW	Usado como <i>background knowledge</i> .	Não	Não
S-Match	Usado como <i>background knowledge</i> e para extração de radicais.	Não	Aponta como trabalhos futuros o uso de <i>feedback</i> do usuário.
ctxMatcher	Usado como <i>background knowledge</i> .	Não	Não
SCARLET	Não	Obtém ontologias por meio do <i>Watson</i> e <i>Swoogle</i>	Não
GeRoMeSuite	Usado para obter sinônimos usados na comparação com a ontologia externa.	Obtém ontologias por meio do Google e <i>Swoogle</i>	Não
ASMOV	Usado como <i>background knowledge</i> .	Não	Não
SemMatcher	Não	Não	Não

Quadro 3.1 - Comparação entre as ferramentas de *ontology matching* analisadas e o *SemMatcher*.

No Quadro 3.1, notamos que o C-SAW, S-Match e ctxMatcher, fazem uso apenas do *WordNet* como conhecimento externo. Apesar de o *WordNet* ser uma boa fonte de conhecimento, há muito mais conhecimento a ser utilizado na Internet. Ontologias que são compartilhadas e atualizadas por várias pessoas expressam um maior valor semântico do que o *WordNet*.

Uma ferramenta que aproveita o potencial da Internet como fonte de conhecimento adicional é o SCARLET. Como foi explicado, o SCARLET faz uso de ontologias encontradas em sites como *Swoogle* e *Watson*. Porém, a comparação usada

por ele é apenas sintática, sem verificar o uso de possíveis sinônimos. O trabalho desenvolvido em [Po and Bergamaschi 2010] sugere o uso do *WordNet* para buscar sinônimos que podem ser usados para melhorar essa comparação.

O *GeRoMeSuite* também acessa ontologias externas para encontrar relacionamentos entre os elementos das ontologias de entrada, assim como o SCARLET, e é uma ferramenta que obteve bons resultados na última edição do OAEI. Além das ferramentas citadas, existem outras não mencionadas como o CLIO [Fagin *et al.* 2009], que assim como outras técnicas de *ontology matching* atuais [Belhajjame *et al.* 2010; Duan *et al.* 2010], tentam aumentar a participação do usuário no processo de *ontology matching*, permitindo ao usuário adicionar ou remover correspondências de acordo com o resultado gerado. Porém, na abordagem usada por eles, esses *feedbacks* obtidos do usuário não são armazenados para posterior uso, desperdiçando conhecimento que poderia ser útil em uma futura operação de *matching*.

No próximo capítulo apresentaremos as mudanças propostas para a nova versão do *SemMatcher*, chamada *SemMatcher+*, incluindo as funcionalidades que foram analisadas nas ferramentas anteriormente descritas.

4. SemMatcher+

Como visto nos capítulos anteriores, muitas são as técnicas usadas para a identificação de correspondências entre ontologias. Entretanto, nenhuma ainda apareceu como a solução definitiva. Os dez desafios da área de *ontology matching* são analisados em [Shvaiko and Euzenat 2008], onde os autores falam sobre cada tendência e o futuro da área. Nosso trabalho tem foco em dois desses desafios: o envolvimento do usuário no processo de *matching* e a busca de novo conhecimento por meio de ontologias disponíveis na Internet.

No primeiro caso, o envolvimento do usuário deve o menos intrusivo possível para o usuário no processo de *matching*. Já para a descoberta de conhecimento em uma fonte externa, algumas das alternativas são: o reuso de correspondências passadas e consultas a ontologias em sites como *Watson* e *Swoogle*.

A ferramenta *SemMatcher* foi estendida para usar técnicas que proporcionam melhorias exatamente nessas áreas, explorando conhecimento adicional disponível na Internet e reusando informações de *feedback* do usuário referentes a processos de *matching* anteriores.

Além disso, as regras para correspondências entre propriedades foram modificadas para analisar o conceito que as contém ao fazer a inferência de relacionamentos. A nova versão será chamada de *SemMatcher+* e, quando falarmos de *SemMatcher*, estaremos nos referindo à versão anterior.

Na Seção 4.1 será apresentada a arquitetura do *SemMatcher+*, contendo módulos referentes a cada uma de suas novas funcionalidades e então detalhamos cada parte da arquitetura. Na Seção 4.2 falamos sobre o *matcher* linguístico-estrutural, e na Seção 4.3 sobre o *matcher* semântico. Em seguida, após a combinação dos *matchers*, na Seção 4.4, mostramos como o histórico é usado para identificar correspondências que não são válidas a partir da avaliação de correspondências anteriores. Por fim, na Seção 4.5, mostramos que, após executado o processo do *matching*, o usuário possui a possibilidade de avaliar as correspondências geradas.

4.1 Arquitetura do SemMatcher+

Para incluir as funcionalidades no *SemMatcher*, uma nova arquitetura é proposta, visando alcançar uma melhor acurácia nos alinhamentos gerados. A Figura 4.1 apresenta a arquitetura do *SemMatcher+*:

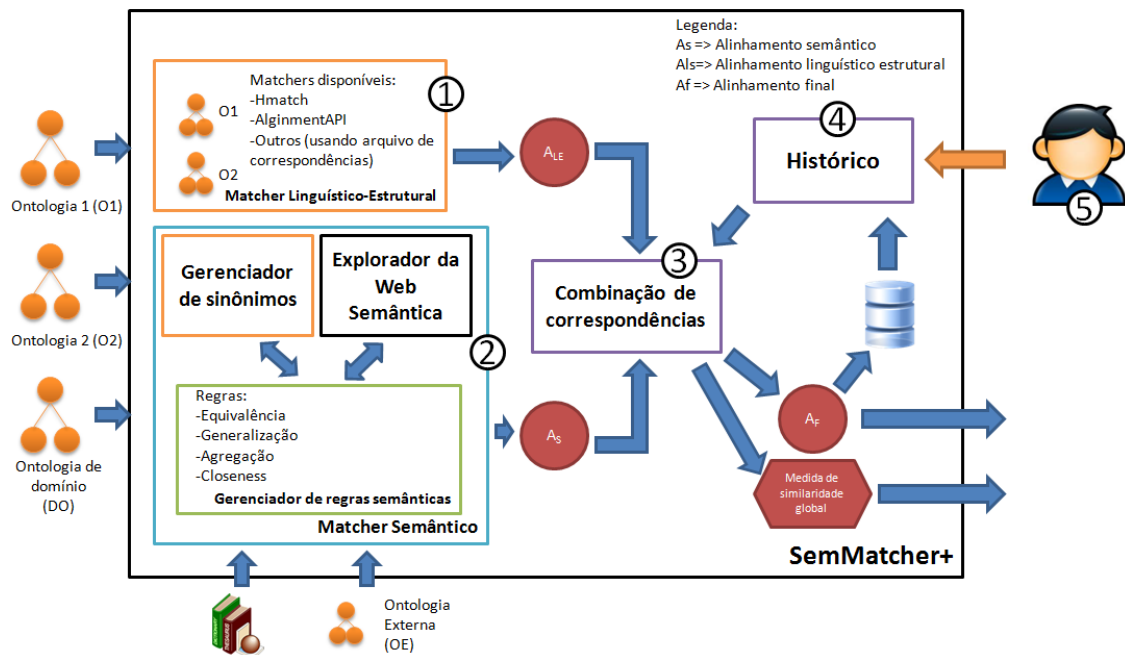


Figura 4.1 – Nova arquitetura do *SemMatcher*

Como na versão anterior (detalhada na Seção 3.9), três ontologias são fornecidas como entrada: as duas ontologias a serem comparadas (O_1 e O_2) e uma ontologia de domínio (DO). O *matcher* linguístico-estrutural (1) é executado gerando o alinhamento A_{LE} . O *matcher* semântico (2) gera o alinhamento A_S com correspondências geradas usando a ontologia de domínio DO como *background knowledge*. Para encontrar os conceitos equivalentes entre as ontologias comparadas (O_1 e O_2) e a ontologia de domínio DO, o *SemMatcher+* utiliza não apenas os nomes dos conceitos mas também seus sinônimos. Além disso, o *matcher* semântico busca conhecimento adicional por meio de uma ontologia externa (OE) disponível na Internet.

Após a execução dos dois *matchers*, fazemos a combinação (3) dos alinhamentos A_{LE} e A_S . Para cada correspondência do alinhamento combinado é verificada sua existência no histórico (4) de correspondências. Se neste histórico a correspondência foi avaliada negativamente por diversas vezes em alinhamentos envolvendo ontologias semelhantes às do processo corrente, a correspondência pode ser eliminada. Após esse processo, o alinhamento final A_F é gerado e armazenado em uma base de dados. O alinhamento A_F contém correspondências descritas no formato de uma 5-tupla (e, e', r, w, n) , onde: e é um elemento da ontologia O_1 , também chamado sujeito, e' é um elemento da ontologia O_2 , também chamado objeto, r é o relacionamento entre e e e' , w é o peso do relacionamento r , e n é o grau de confiança do relacionamento r . Este último componente, o grau de confiança, foi incluído no *SemMatcher+* para representar

o quão confiável é uma determinada correspondência. A medida de similaridade global entre as duas ontologias também é calculada e armazenada no histórico

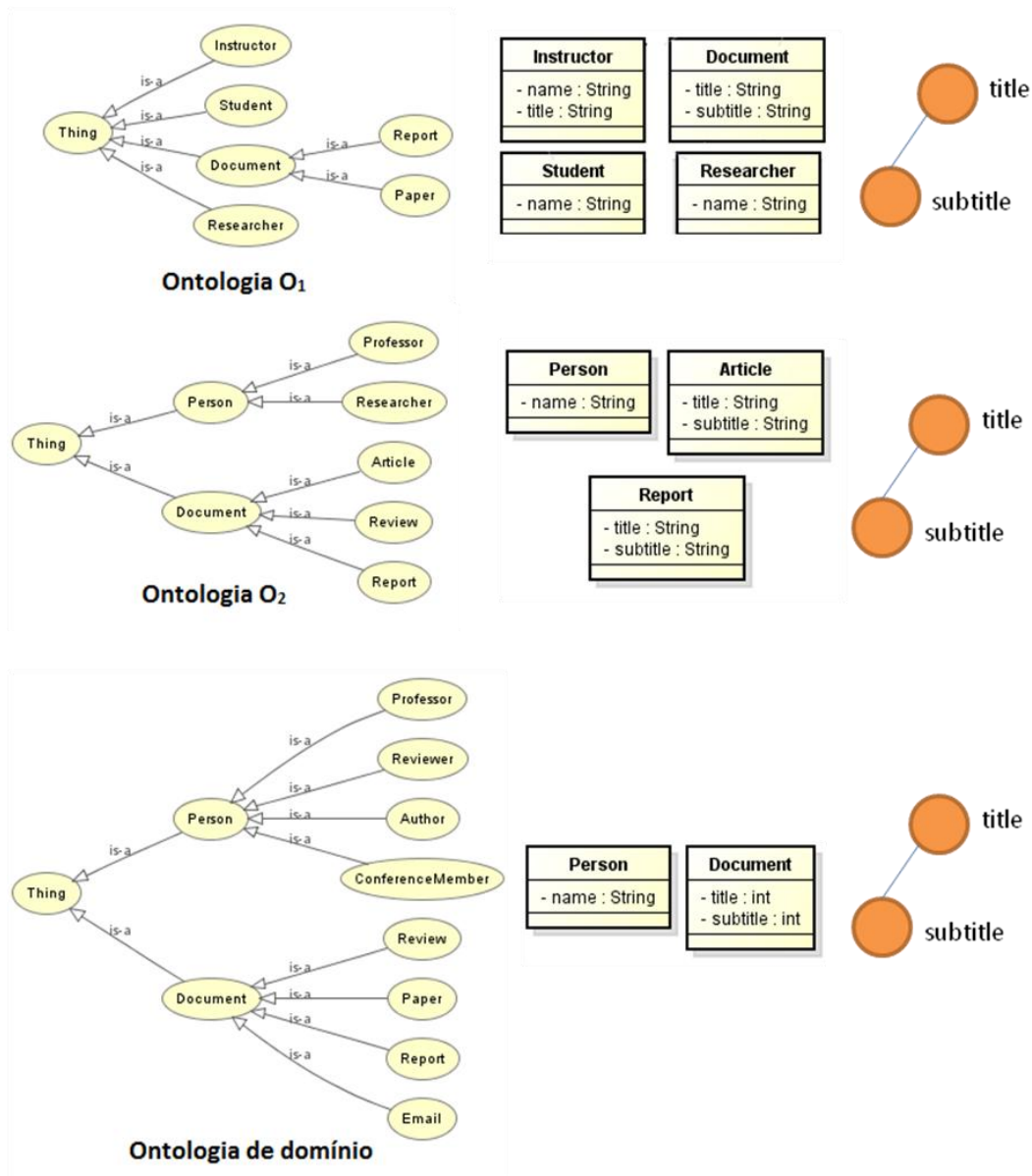


Figura 4.2 - Ontologias usadas para descrever o processo do *SemMatcher+*

No *SemMatcher+*, não existe a fase de *Seleção de Correspondências* da versão anterior, pois foi verificado que a seleção eliminava muitas correspondências úteis do alinhamento final durante o processo de escolha das melhores correspondências. As correspondências que eram filtradas eram relevantes, pois foram inferidas a partir de relacionamentos na ontologia de domínio fornecida. Por este motivo, a escolha dos melhores alinhamentos é usada apenas para o cálculo da similaridade global, mas não influi no alinhamento final A_F gerado pelo *SemMatcher+*. A seleção ainda é feita para o cálculo porque as fórmulas usadas fazem necessário que haja apenas um valor de similaridade para cada conceito.

Após apresentar o resultado final (alinhamento A_F e medida de similaridade), será dada ao usuário a possibilidade de informar que correspondências não estão corretas naquele alinhamento. Esta operação causa uma atualização no histórico e o grau de confiança associado à correspondência daquele alinhamento específico será atualizado para zero. Nas próximas seções apresentaremos o funcionamento de cada módulo em detalhes. Para isso usaremos um cenário com ontologias pequenas para ilustrar como acontece o processo. As três ontologias de entrada são mostradas na Figura 4.2. Estas ontologias representam um cenário sobre o domínio de educação. Não foram escolhidas ontologias grandes para facilitar a demonstração de toda a execução. A Figura 4.2 mostra para cada ontologia, a hierarquia de conceitos, as propriedades de cada conceito, e por fim a hierarquia entre propriedades da ontologia.

4.2 *Matcher* Linguístico-Estrutural

Na primeira fase da execução do *SemMatcher+*, qualquer ferramenta de *ontology matching* que gere alinhamentos no formato definido pela OAEI [Euzenat et al. 2005; OAEI 2011] pode ser usada, sendo ela linguística e/ou estrutural. Estas ferramentas geralmente são híbridas, ou seja, combinam algoritmos de *matching* mais simples com técnicas linguísticas ou estruturais.

Como na versão anterior, é possível usar as ferramentas AlignmentAPI [Euzenat 2004] e H-match [Castano et al. 2003] através do próprio *SemMatcher+* ou qualquer ferramenta que gere alinhamentos descritos no formato definido pela OAEI pode ser usada. Ao executar uma das ferramentas, ou utilizando o alinhamento fornecido no formato definido pela OAEI, temos como resultado o alinhamento linguístico-estrutural A_{LE} . No nosso exemplo, obtemos o alinhamento A_{LE} mostrado no Quadro 4.1. As correspondências geradas pela ferramenta externa são geradas no formato 4-tupla. Como consideramos confiáveis as correspondências geradas pela ferramenta linguístico-estrutural usada no processo, formamos uma 5-tupla atribuindo 1.0 como grau de confiança.

Sujeito	Objeto	Relacionamento	Peso	Grau de confiança
Report	Report	Equivalente	1.0	1.0
Document	Document	Equivalente	1.0	1.0
Researcher	Researcher	Equivalente	1.0	1.0
Instructor.name	Person.name	Equivalente	1.0	1.0
Researcher.name	Person.name	Equivalente	1.0	1.0
Student.name	Person.name	Equivalente	1.0	1.0
Instructor.title	Article.title	Equivalente	1.0	1.0
Instructor.title	Report.title	Equivalente	1.0	1.0
Document.title	Article.title	Equivalente	1.0	1.0
Document.title	Report.title	Equivalente	1.0	1.0
Document.subtitle	Report.subtitle	Equivalente	1.0	1.0
Document.subtitle	Article.subtitle	Equivalente	1.0	1.0

Quadro 4.1 - Alinhamento A_{LE} gerado.

4.3 Execução do *Matcher* Semântico

Para enriquecer com semântica as correspondências geradas na fase anterior, executamos o *matcher* semântico, que gera o alinhamento A_S . Posteriormente, fazemos uma combinação dos alinhamentos A_{LE} e A_S gerados pelas duas fases. Nas próximas seções, será detalhada cada uma das novas funcionalidades do *matcher* semântico. Na próxima seção, veremos a mudança nas regras que tratam das propriedades dos conceitos. Na Seção 4.3.2, veremos como sinônimos são usados para encontrar mais conceitos e propriedades equivalentes na ontologia de domínio, possibilitando inferir mais correspondências. O processo interno do *matcher* semântico foi modificado, e será apresentado na Seção 4.3.3.

4.3.1 Regras de Propriedades

O *SemMatcher* executa 7 regras semânticas [Souza 2009], como descrito na Seção 3.9. Três destas regras tratam de conceitos e propriedades. Para o *SemMatcher+*, estas 3 regras foram reformuladas, especificamente as que referem-se aos relacionamentos de equivalência, generalização e especialização. Uma correspondência de equivalência entre propriedades indica que elas representam a mesma propriedade de um conceito no mundo real. Na abordagem do *SemMatcher+*, se duas propriedades (uma em O_1 e outra em O_2) referem-se a uma mesma propriedade na ontologia de domínio, e seus respectivos conceitos têm um como ancestral do outro (em outras palavras, se um conceito tem o outro como superconceito em um ou mais níveis acima na hierarquia). A verificação de conceito não apenas no caso de serem iguais é justificada pelo fato de que um subconceito tem propriedades comuns do conceito pai. Nas ontologias O_1 e O_2 , temos, por exemplo, que *Document.title* é equivalente a *Article.title* pois além da propriedade ser semelhante, os conceitos estão na mesma hierarquia. Por este mesmo

motivo, *Instructor.title* não possui nenhuma correspondência com *Article.title* ou *Report.title*, pois *Instructor* não está na mesma hierarquia que *Article* ou *Report*. A regra de equivalência é definida a seguir.

Definição 8 – Regra *isEquivalentPropertyTo*: uma propriedade $O_1:x.p_1$ é equivalente a $O_2:y.p_2$ se $O_1:x.p_1 \equiv DO:k.p_3$ e $O_2:y.p_2 \equiv DO:z.p_3$ e ($DO:k \equiv DO:z$ ou $DO:k$ tem $DO:z$ como superconceito em um ou mais níveis acima ou $DO:z$ tem $DO:k$ como superconceito em um ou mais níveis acima, sendo qualquer um dos conceitos destacados na ontologia de domínio DO da Figura 4.3). Esta correspondência é representada por $O_1:x.p_1 \equiv O_2:y.p_2$.

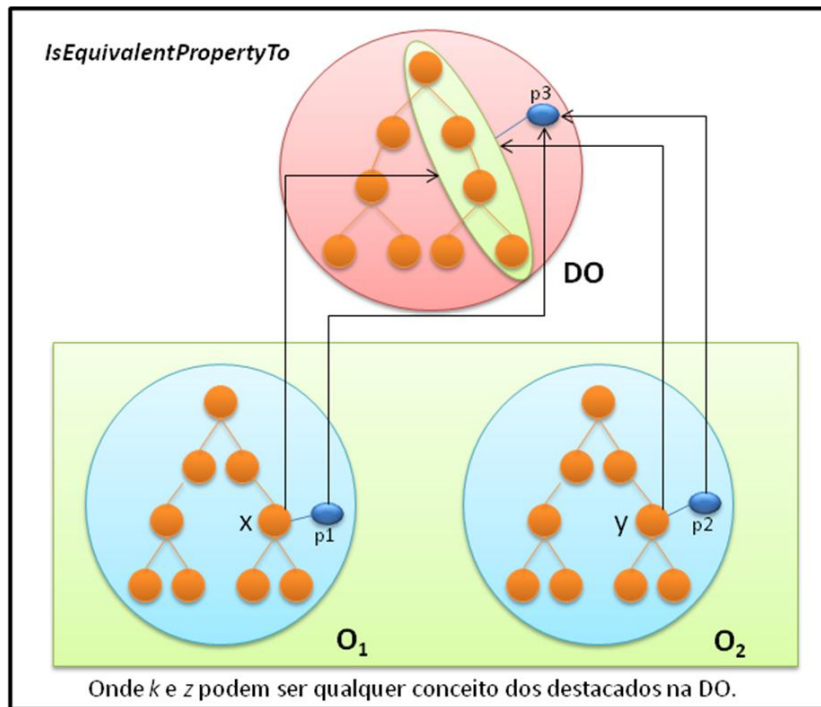


Figura 4.3 - Relacionamento de equivalência entre propriedades

Também foram modificadas as regras para correspondências de generalização e especialização. Se uma propriedade p_1 em um conceito x em O_1 é idêntica a uma propriedade p_3 em um conceito k na DO, e outra propriedade p_2 em um conceito y na O_2 é idêntica a p_4 (superpropriedade de p_3) no conceito z da DO. Além disso, se os conceitos k e z têm um como ancestral do outro, em outras palavras, se k tem z como superconceito em um ou mais níveis acima na hierarquia (na Figura 4.4 vemos que k e z podem ser qualquer um dos conceitos destacados na ontologia de domínio DO). A propriedade $x.p_1$ é subpropriedade de $y.p_2$. Considerando as ontologias de exemplo, inferimos que *Document.subtitle* é subpropriedade de *Article.title* pois *subtitle* é subpropriedade de *title*, e *Document* e *Article* estão na mesma hierarquia. A regra de subpropriedade é definida a seguir.

Definição 9 – Regra *isSubPropertyOf*: um elemento $O_1:x.p_1$ é subpropriedade de $O_2:y.p_2$ se $O_1:x.p_1 \equiv DO:k.p_3$ e $O_2:y.p_2 \equiv DO:z.p_4$ e $DO:k.p_3 \subseteq DO:z.p_4$ e ($DO:k \equiv DO:z$ ou $DO:k$ tem $DO:z$ como superconceito em um ou mais níveis acima ou $DO:z$ tem $DO:k$ como superconceito em um ou mais níveis acima, sendo qualquer um dos conceitos destacados na ontologia de domínio DO da Figura 4.4). Esta correspondência é representada por $O_1:x.p_1 \sqsubseteq O_2:y.p_2$.

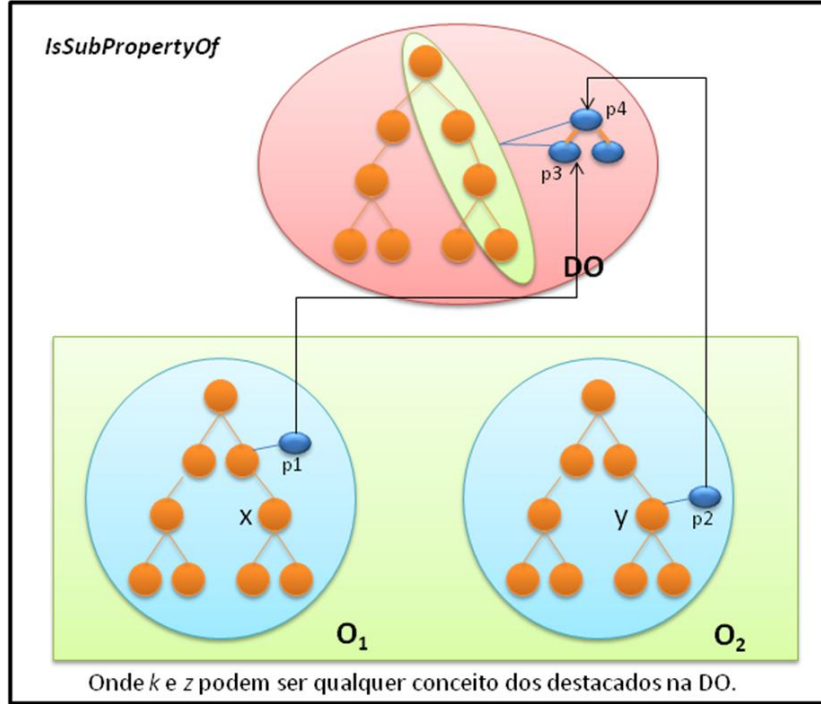


Figura 4.4 - Relacionamento de especialização entre propriedades

No caso da generalização, se uma propriedade p_1 em um conceito x em O_1 é idêntica a uma propriedade p_3 em um conceito k na DO, e outra propriedade p_2 em um conceito y na O_2 é idêntica a uma propriedade p_4 (subpropriedade de p_3) de z da DO. Além disso, se os conceitos k e z têm um como ancestral do outro, em outras palavras, se k tem z como superconceito em um ou mais níveis (na Figura 4.5 vemos que k e z podem ser qualquer um dos conceitos destacados na ontologia de domínio DO). A propriedade $x.p_1$ é superpropriedade de $y.p_2$. Considerando as ontologias de exemplo, inferimos que *Document.title* é superpropriedade de *Report.subtitle* pois *title* é superpropriedade de *subtitle*, e *Document* e *Report* estão na mesma hierarquia.

Definição 10 – Regra *isSuperPropertyOf*: um elemento $O_1:x.p_1$ é superpropriedade de $O_2:y.p_2$ se $O_1:x.p_1 \equiv DO:k.p_3$ e $O_2:y.p_2 \equiv DO:z.p_4$ e $DO:k.p_3 \supseteq DO:z.p_4$ e ($DO:k \equiv DO:z$ ou $DO:k$ tem $DO:z$ como superconceito em um ou mais níveis acima ou $DO:z$ tem $DO:k$ como superconceito em um ou mais níveis acima, sendo

qualquer um dos conceitos destacados na ontologia de domínio DO da Figura 4.5). Esta correspondência é representada por $O_1:x.p_1 \rightarrow O_2:y.p_2$.

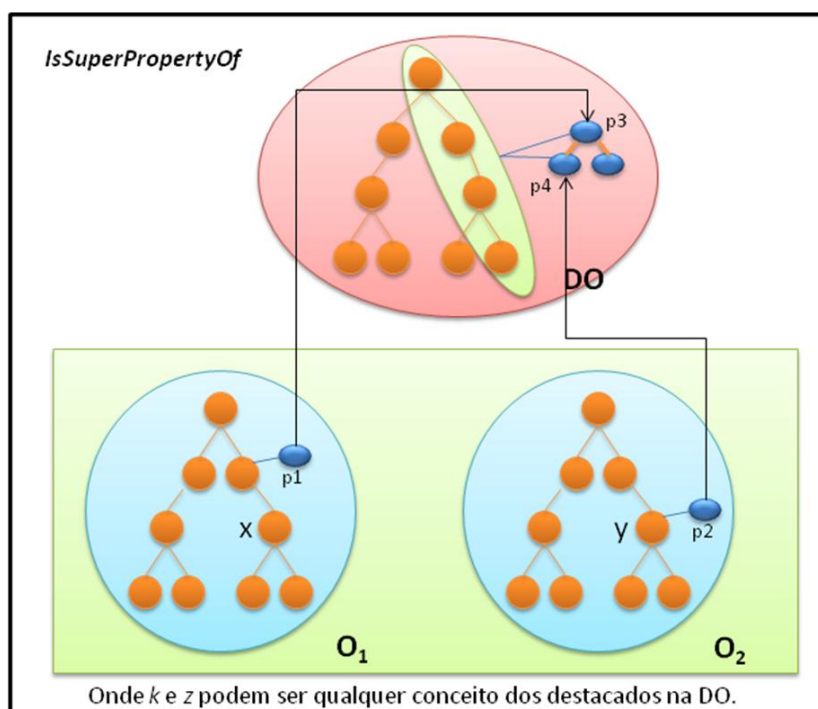


Figura 4.5 - Relacionamento de generalização entre propriedades

4.3.2 Gerenciador de Sinônimos

O *matcher* semântico encontra conceitos e propriedades equivalentes entre as ontologias comparadas e a ontologia de domínio DO para então inferir as correspondências semânticas. No *SemMatcher*, para encontrar os conceitos e propriedades correspondentes na ontologia de domínio, a comparação é feita apenas verificando se o nome dos conceitos ou propriedades são iguais sintaticamente. Neste trabalho, também verificamos se o conceito é igual a algum de seus sinônimos, consultando um dicionário de sinônimos. Por exemplo, na ontologia O_1 temos o conceito *Instructor*. Este conceito não está presente na ontologia de domínio, porém, ao checar seus sinônimos em um dicionário de sinônimos, encontramos o conceito *Professor*. Como na O_2 *Professor* se refere ao mesmo *Professor* da ontologia de domínio que *Instructor* da O_1 se refere, inferimos que *Instructor* é equivalente a *Professor*.

4.3.3 Processos do *Matcher* Semântico

O funcionamento interno do *matcher* semântico também passou por mudanças. Na Figura 4.6 vemos o novo processo de execução do *matcher* semântico, com destaque para os módulos a seguir: Gerenciador de regras semânticas, Explorador da Web Semântica e Combinação de correspondências.

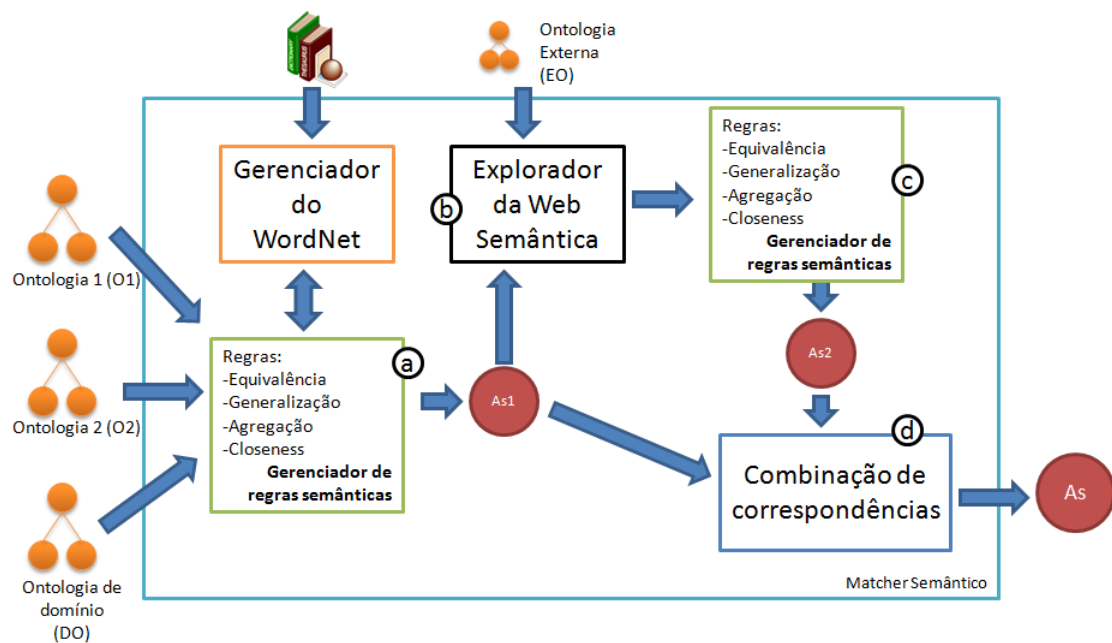


Figura 4.6 – Processo do *matcher* semântico

Sujeito	Objeto	Relacionamento	Peso	Grau de confiança
Report	Report	<i>isEquivalentTo</i>	1.0	1.0
Document	Document	<i>isEquivalentTo</i>	1.0	1.0
Report	Document	<i>isSubConceptOf</i>	0.8	1.0
Paper	Document	<i>isSubConceptOf</i>	0.8	1.0
Document	Review	<i>isSuperConceptOf</i>	0.8	1.0
Document	Report	<i>isSuperConceptOf</i>	0.8	1.0
Report	Review	<i>isCloseTo</i>	0.7	1.0
Paper	Report	<i>isCloseTo</i>	0.7	1.0
Instructor.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Researcher.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Student.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Article.title	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Report.title	<i>isEquivalentTo</i>	1.0	1.0
Document.subtitle	Report.subtitle	<i>isEquivalentTo</i>	1.0	1.0
Document.subtitle	Article.subtitle	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Report.subtitle	<i>isSubPropertyOf</i>	0.8	1.0
Document.title	Article.subtitle	<i>isSubPropertyOf</i>	0.8	1.0
Document.subtitle	Article.title	<i>isSuperPropertyOf</i>	0.8	1.0
Document.subtitle	Report.title	<i>isSuperPropertyOf</i>	0.8	1.0

Quadro 4.2 - Alinhamento A_{S1} gerado.

(a) Primeira execução do Gerenciador de regras semânticas.

O *matcher* semântico recebe as três ontologias passadas como parâmetro para o *SemMatcher+* e faz a execução das regras semânticas pela primeira vez, gerando um alinhamento inicial A_{S1} . Os alinhamentos aqui gerados contêm correspondências no formato de uma 5-tupla como descritos na Seção 4.1. Nesta primeira execução do gerenciador, o grau de confiança atribuído é 1 para todas as correspondências de A_{S1} , pelo fato de os relacionamentos terem sido inferidos a partir de uma ontologia de

domínio passada como parâmetro. O valor do grau de confiança pode diminuir em fases subsequentes do processo, como será explicado posteriormente. Após a execução do gerenciador de regras, temos o alinhamento A_{S1} mostrado no Quadro 4.2.

(b) Explorador da Web Semântica

As correspondências geradas na primeira execução do gerenciador (A_{S1}) são passadas para o módulo Explorador da Web Semântica. A sua função é identificar os conceitos para os quais não foram encontradas correspondências, e usá-los como termos para realizar uma busca por ontologias externas que possam ter conhecimento adicional sobre eles. Para isso, são verificados quais conceitos das ontologias O_1 e O_2 não estão presentes nem no sujeito, nem no objeto das correspondências geradas.

Esses conceitos são usados para fazer uma busca por ontologias externas disponíveis na Internet. No nosso exemplo, pelo alinhamento A_{S1} vemos que na ontologia O_1 os conceitos *Student*, *Instructor* e *Researcher* não estão presentes em correspondências. Da mesma forma, em O_2 , os conceitos *Person*, *Professor*, *Article* e *Researcher* não estão presentes nas correspondências de A_{S1} . O número máximo de conceitos usados na busca é configurável. Por padrão, metade dos conceitos é de O_1 e a outra metade de O_2 . A escolha foi feita desta forma para encontrar uma ontologia que contenha conceitos referentes às duas ontologias, aumentando a chance de encontrar um relacionamento, pois a ontologia encontrada teria que ter conceitos das duas ontologias comparadas. Um número muito grande de conceitos na busca pode não retornar resultados por ser um critério de busca muito restritivo. Da mesma forma, pode-se configurar se deseja-se dar preferência a conceitos em um nível mais alto na hierarquia das ontologias comparadas (O_1 e O_2), ou mais baixo, de acordo com a preferência de buscar conhecimento mais genérico ou mais específico, respectivamente. Usamos o limite de quatro conceitos e preferência para conceitos mais acima na hierarquia neste exemplo, desta forma, da ontologia O_1 temos *Student* e *Instructor* e da ontologia O_2 temos *Person* e *Professor*. As quatro palavras-chave são então submetidas a um dos sistemas de busca de ontologias (*Watson* ou *Swoogle*) na Internet. Feita a busca, a ontologia melhor colocada (primeira ontologia retornada) no resultado é selecionada para ser usada como ontologia externa para uma nova execução do gerenciador de regras semânticas.

(c) Segunda execução do gerenciador de regras semânticas.

As duas ontologias O_1 e O_2 são passadas para o gerenciador de regras semânticas, além da ontologia externa encontrada pelo explorador da Web Semântica.

Um novo alinhamento A_{S2} é gerado e atribuímos a cada correspondência o grau de confiança igual 0.8. Este valor foi atribuído arbitrariamente por considerarmos que a ontologia encontrada na Internet é menos confiável que a ontologia provida pelo usuário que foi usada na primeira execução do gerenciador de regras semânticas. No nosso exemplo, apenas duas correspondências foram encontradas, o alinhamento A_{S2} completo é mostrado no Quadro 4.3. A ontologia encontrada neste caso não contém propriedades, porém se uma ontologia com propriedades fosse encontrada, novas correspondências entre propriedades também poderiam ser encontradas.

Sujeito	Objeto	Relacionamento	Peso	Grau de confiança
Document	Document	<i>isEquivalentTo</i>	1.0	0.8
Student	Person	<i>isSubConceptOf</i>	0.8	0.8

Quadro 4.3 - Alinhamento A_{S2} gerado.

Sujeito	Objeto	Relacionamento	Peso	Grau de confiança
Report	Report	<i>isEquivalentTo</i>	1.0	1.0
Document	Document	<i>isEquivalentTo</i>	1.0	1.0
Report	Document	<i>isSubConceptOf</i>	0.8	1.0
Paper	Document	<i>isSubConceptOf</i>	0.8	1.0
Document	Review	<i>isSuperConceptOf</i>	0.8	1.0
Document	Report	<i>isSuperConceptOf</i>	0.8	1.0
Report	Review	<i>isCloseTo</i>	0.7	1.0
Paper	Report	<i>isCloseTo</i>	0.7	1.0
Instructor.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Researcher.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Student.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Article.title	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Report.title	<i>isEquivalentTo</i>	1.0	1.0
Document.subtitle	Report.subtitle	<i>isEquivalentTo</i>	1.0	1.0
Document.subtitle	Article.subtitle	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Report.subtitle	<i>isSubPropertyOf</i>	0.8	1.0
Document.title	Article.subtitle	<i>isSubPropertyOf</i>	0.8	1.0
Document.subtitle	Article.title	<i>isSuperPropertyOf</i>	0.8	1.0
Document.subtitle	Report.title	<i>isSuperPropertyOf</i>	0.8	1.0
Student	Person	<i>isSubConceptOf</i>	0.8	0.8

Quadro 4.4 - Alinhamento A_S gerado.

(d) Enriquecimento com as correspondências externas

Os dois alinhamentos A_{S1} e A_{S2} são unidos. Quando há correspondências repetidas (o sujeito e objeto são idênticos), dá-se preferência à correspondência que apresenta um maior grau de confiança, ou seja, aquela encontrada a partir da ontologia de domínio passada como parâmetro (primeira execução do gerenciador de regras

semânticas), pertencente a A_{S1} . No nosso exemplo, a correspondência de equivalência entre $O_1.Document$ e $O_2.Document$ está presente tanto em A_{S1} como em A_{S2} . A correspondência presente em A_{S2} é descartada por ter grau de confiança menor que a de A_{S1} . Da união desses alinhamentos temos o alinhamento A_S mostrado no Quadro 4.4.

Sujeito	Objeto	Relacionamento	Peso	Grau de confiança
Report	Report	<i>isEquivalentTo</i>	1.0	1.0
Document	Document	<i>isEquivalentTo</i>	1.0	1.0
Report	Document	<i>isSubConceptOf</i>	0.6	1.0
Paper	Document	<i>isSubConceptOf</i>	0.48	1.0
Student	Person	<i>isSubConceptOf</i>	0.48	0.8
Document	Review	<i>isSuperConceptOf</i>	0.48	1.0
Document	Report	<i>isSuperConceptOf</i>	0.48	1.0
Report	Review	<i>isCloseTo</i>	0.42	1.0
Paper	Report	<i>isCloseTo</i>	0.42	1.0
Instructor.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Researcher.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Student.name	Person.name	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Article.title	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Report.title	<i>isEquivalentTo</i>	1.0	1.0
Document.subtitle	Report.subtitle	<i>isEquivalentTo</i>	1.0	1.0
Document.subtitle	Article.subtitle	<i>isEquivalentTo</i>	1.0	1.0
Document.title	Report.subtitle	<i>isSubPropertyOf</i>	0.48	1.0
Document.title	Article.subtitle	<i>isSubPropertyOf</i>	0.48	1.0
Document.subtitle	Article.title	<i>isSuperPropertyOf</i>	0.48	1.0
Document.subtitle	Report.title	<i>isSuperPropertyOf</i>	0.48	1.0
Researcher	Researcher	-	0.4	1.0
Instructor.title	Article.title	-	0.4	1.0
Instructor.title	Report.title	-	0.4	1.0

Quadro 4.5 - Alinhamento A_F gerado.

4.4 Histórico

Após a execução dos *matchers* linguístico-estrutural e semântico, temos dois alinhamentos gerados, o A_{LE} e o A_S . No passo 3 da Figura 4.1, combinamos esses dois alinhamentos calculando uma média ponderada entre os pesos dos relacionamentos das correspondências. Para cada correspondência no alinhamento A_{LE} , é procurada a sua equivalente (de sujeito e objeto igual) no A_S e, se encontrada, é feita uma média ponderada entre os seus pesos como no *SemMatcher*. Quando não há uma correspondência equivalente em A_{LE} ou A_S , o peso da correspondência atribuído é zero para aquela correspondência. É criado um novo alinhamento após esses cálculos

chamado de alinhamento combinado A_F mostrado no Quadro 4.5, que é então passado para o Módulo de Histórico de Alinhamentos.

O *SemMatcher+* permite a avaliação de correspondências pelo usuário. A invalidação de uma correspondência provoca uma diminuição do grau de confiança da correspondência. Neste momento (Passo 4 da Figura 4.1), o alinhamento A_F é passado para o módulo de histórico. Avaliações anteriores feitas de correspondências pertencentes a alinhamentos que envolvam ontologias semelhantes (mesmo domínio) as atuais são analisadas e usadas para remover correspondências que não são válidas no alinhamento A_F corrente. Para selecionar os alinhamentos semelhantes do histórico que terão avaliações a serem verificadas, são analisados apenas os alinhamentos anteriores do *SemMatcher+* que possuem ontologias semelhantes às da operação de *matching* corrente.

Considerando as duas ontologias comparadas O_1 e O_2 do nosso exemplo, vamos exemplificar como ocorre essa escolha dos alinhamentos anteriores. Supondo que, neste ponto da execução, existam três alinhamentos anteriores armazenados como mostrado na Figura 4.7, sendo eles:

- 1) Alinhamento A_1 , entre as ontologias O_3 e O_4 , com medida de similaridade global igual a 0.6;
- 2) Alinhamento A_2 , entre as ontologias O_1 e O_3 , com medida de similaridade global igual a 0.8; e
- 3) Alinhamento A_3 , entre as ontologias O_2 e O_4 , com medida de similaridade global igual a 0.9;

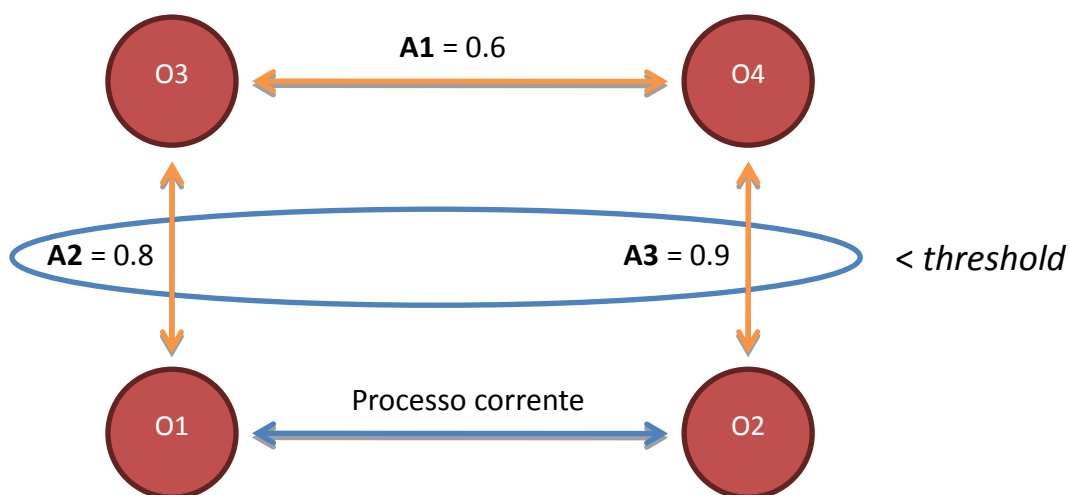


Figura 4.7 - Identificação de processos de *matching* anteriores semelhantes ao atual.

Para identificar alinhamentos anteriores que possuem ontologias semelhantes, procuramos no histórico as ontologias mais semelhantes a O_1 e O_2 baseado na medida

de similaridade global. Encontradas as ontologias mais semelhantes a O_1 e O_2 , procuramos alinhamentos entre essas ontologias. No caso da Figura 4.7, a partir de O_1 , uma das ontologias mais semelhantes encontrada foi O_3 , baseado na medida de similaridade global armazenada com o alinhamento A_2 , com valor de 0.8. Da mesma forma, a partir de O_2 , uma das ontologias mais semelhantes encontradas foi O_4 , baseado na medida de similaridade global armazenada com o alinhamento A_3 . Em ambos os casos consideramos que eles são semelhantes se tiverem medida de similaridade acima de um limiar configurável, no nosso exemplo usamos o valor 0.7. A partir da informação que O_3 e O_4 são semelhantes às ontologias correntes, encontramos um alinhamento existente entre elas, A_1 .

Obtidos os alinhamentos anteriores que envolvem ontologias semelhantes às ontologias do alinhamento corrente, para cada correspondência do alinhamento corrente, é atualizado o grau de confiança da correspondência atual, levando em consideração as correspondências anteriores. O novo grau de confiança para a correspondência do alinhamento corrente será a média ponderada entre o valor de confiança atual, e o valor da média dos graus de confiança dos alinhamentos anteriores, onde os pesos da média ponderada são configuráveis. A fórmula para o cálculo do novo grau de confiança é mostrado na Equação 4.1.

$$N(\text{novo grau de confiança}) = P_{\text{anteriores}} * (\text{Média dos graus de confiança anteriores para determinada correspondência}) + P_{\text{corrente}} * (\text{Grau de confiança da correspondência corrente})$$

(4.1)

Correspondências com um grau de confiança próximo a zero (valor configurável) são removidas, e as correspondências restantes formam o alinhamento final A_F . Se nenhum alinhamento anterior foi considerado semelhante ao atual, nada é alterado e o alinhamento de entrada é igual ao alinhamento final. O alinhamento final é então armazenado para posterior uso em uma nova execução do *SemMatcher*.

4.5 Feedback do Usuário

Ao final do processo de *matching*, o *SemMatcher+* exibe o alinhamento final A_F para o usuário. Neste ponto, o usuário pode avaliar o alinhamento gerado, indicando correspondências inválidas. As correspondências do alinhamento atual, marcadas como inválidas pelo usuário, têm seu grau de confiança atualizado para zero para o alinhamento atual, o que impacta no cálculo do grau de confiança de novos alinhamentos gerados na fase de histórico. A avaliação pode ser feita usando a interface

do *SemMatcher+*, onde o usuário tem a possibilidade de avaliar cada uma das correspondências quando o alinhamento final é apresentado.

4.6 Considerações Finais

Neste capítulo apresentamos as novas funcionalidades adicionadas ao *SemMatcher+*, incluindo a descoberta de novas correspondências, o aumento do grau de confiança de correspondências existentes, e a filtragem de correspondências inválidas baseadas no *feedback* do usuário.

O *SemMatcher+* faz uso de ontologias de domínio passadas como parâmetro, para obter relacionamentos semânticos. Na nossa nova abordagem, ontologias externas também são pesquisadas na Internet, usando os sites *Watson* e *Swoogle*, para inferir mais relacionamentos semânticos. Além disso, na descoberta de relacionamentos semânticos, não apenas o nome exato do conceito é considerado, como também seus sinônimos. Ao final do processo, as correspondências podem ser invalidadas pelo usuário caso ele as ache incorretas. No próximo capítulo vamos mostrar em detalhes a implementação das mudanças propostas e os resultados dos experimentos realizados sobre o *SemMatcher+*.

5. Implementação e Experimentos

Neste capítulo entraremos em mais detalhes sobre a implementação das novas funcionalidades propostas para o *SemMatcher+*. Na Seção 5.1, apresentamos as bibliotecas utilizadas e como foi feita a escolha delas. Após falarmos sobre a implementação, na Seção 5.2 mostraremos os experimentos realizados.

5.1 Tecnologias utilizadas

Para a implementação do *SemMatcher+*, foi utilizada a linguagem de programação Java. O estudo de algumas bibliotecas e ferramentas foi necessário. A seguir veremos como foram implementadas cada uma das novas funcionalidades do *SemMatcher+*.

5.1.1 Bibliotecas para o *WordNet*

Para o uso de sinônimos na identificação de correspondências entre as ontologias e a ontologia de domínio, foi necessário o estudo do *WordNet* e de bibliotecas que possibilitassem o seu uso junto com Java. As bibliotecas analisadas foram:

- JWNL (*Java WordNet Library*) [JWNL 2011]

JWNL é uma biblioteca *open-source* disponível no *sourceforge*. Teve sua última versão lançada em 2008 e, apesar da implementação de uma nova versão ter começado, esta nunca foi lançada. A JWNL possui uma documentação escassa, além disso, o código necessário para realizar as operações era complexo.

- JAWS (*Java API for WordNet Searching*) [JAWS 2011]

O JAWS é uma biblioteca *open-source* e teve sua última versão lançada em 2009. Possui API bem escrita, mas apresenta apenas um exemplo de código. Alguns problemas de configuração também ocorreram, tornando o seu uso problemático.

- JWI (*Java WordNet Interface*) [JWI 2011]

A JWI é uma biblioteca desenvolvida pelo *Massachusetts Institute of Technology* (MIT), sua versão atual é de 2008, mas possui uma boa documentação, tanto em código como em tutoriais. Outra vantagem é a possibilidade de uso em projetos não comerciais.

Por todas as qualidades descritas, a JWI foi a escolhida para ser usada no acesso ao *WordNet*. A integração com o *SemMatcher* foi fácil e o desempenho bom.

5.1.2 Engenhos de busca de ontologias

Ainda em relação ao *matcher* semântico, quando é feita a exploração de ontologias na Internet, foram usados dois engenhos de busca de ontologias bem conhecidos: o *Swoogle* [Swoogle 2011] e o *Watson* [d'Aquin *et al.* 2007].

- *Swoogle*

O *Swoogle* é um sistema de busca de ontologia que se baseia em palavra-chave utilizando a biblioteca *Lucene* [Lucene 2011] para criar a indexação de ontologias. Não há raciocínio sobre as palavras usadas na busca, apenas uma busca por palavra-chave sobre ontologias indexadas. Para seu uso no *SemMatcher*, foi usado a interface REST (*Representational State Transfer*), que possibilitou a comunicação entre o *SemMatcher* e sistema de busca do *Swoogle*. Existem algumas limitações no seu uso normal, pois é necessário passar uma chave quando uma consulta é feita, e a chave fornecida pelo site limita o número de resultados. Um contato foi feito com os desenvolvedores que forneceram uma chave sem limitações de uso.

- *Watson*

O *Watson* é um sistema de busca de ontologias mais elaborado. Além de indexar as ontologias, o sistema raciocina sobre os conceitos e relacionamentos contidos nelas. Não apenas os relacionamentos explícitos são usados, mas os relacionamentos implícitos também são inferidos. O processo de indexação das ontologias no *Watson* consiste basicamente de três passos mostrados na Figura 5.1.

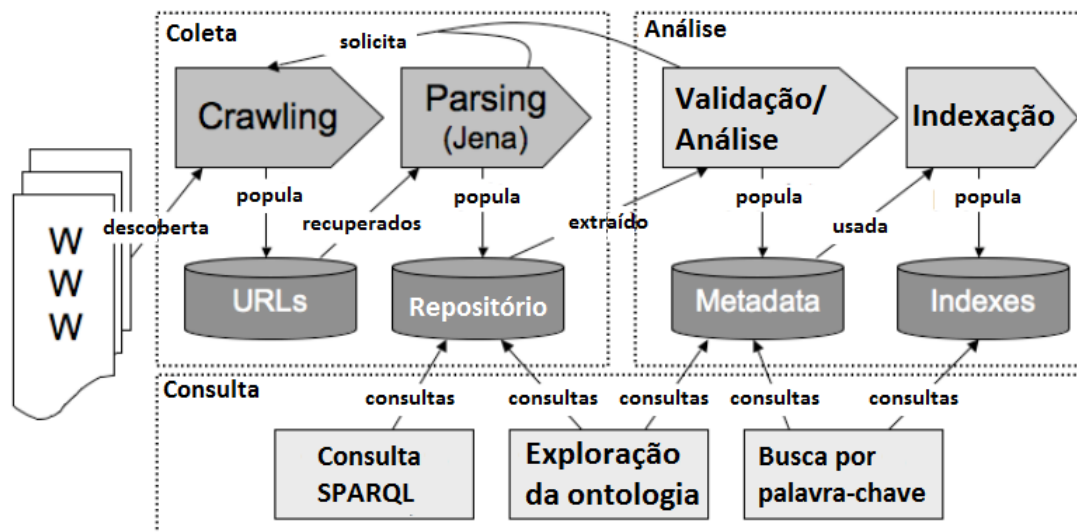


Figura 5.1 – Arquitetura do Watson [d’Aquin *et al.* 2007]

A primeira fase é de coleta, várias fontes são consultadas, desde o próprio Swoogle até o Google (usando o filtro “*filetype:owl*”). Ainda nesta fase, o Jena é usado para filtrar quais documentos encontrados são realmente ontologias válidas.

As ontologias selecionadas são passadas para a fase de Análise, onde o conteúdo de cada ontologia é analisado, incluindo conceitos, propriedades, indivíduos e literais. Além disso, são calculadas medidas que verificam a qualidade do conhecimento expresso na ontologia. Esses dados são usados para indexar as ontologias. Antes da próxima fase, ainda é verificada a existência de relacionamentos entre diferentes ontologias.

Na última fase, está localizada a interface Web do *Watson*, que possibilita a busca de ontologias por palavras-chave até a realização de consultas mais complexas usando SPARQL. É possível navegar pelos conceitos e encontrar uma descrição sobre cada conceito (obtida na própria ontologia ou em outras que contenham o mesmo conceito). Além da interface Web, é possível fazer consultas por meio da *Watson API*, que faz acesso via *web service* ao conteúdo do *Watson*. No *SemMatcher+*, as consultas ao *Watson* são feitas utilizando essa API.

5.1.3 Controle de histórico das correspondências

O *SemMatcher+* gerencia o armazenamento de correspondências em um banco de dados próprio, independente do armazenamento fornecido pelo Jena utilizado no *SemMatcher*. O SGBD utilizado para implementação foi o *MySQL*. A Figura 5.2 ilustra

o esquema relacional usado para armazenamento dos alinhamentos e de suas correspondências.

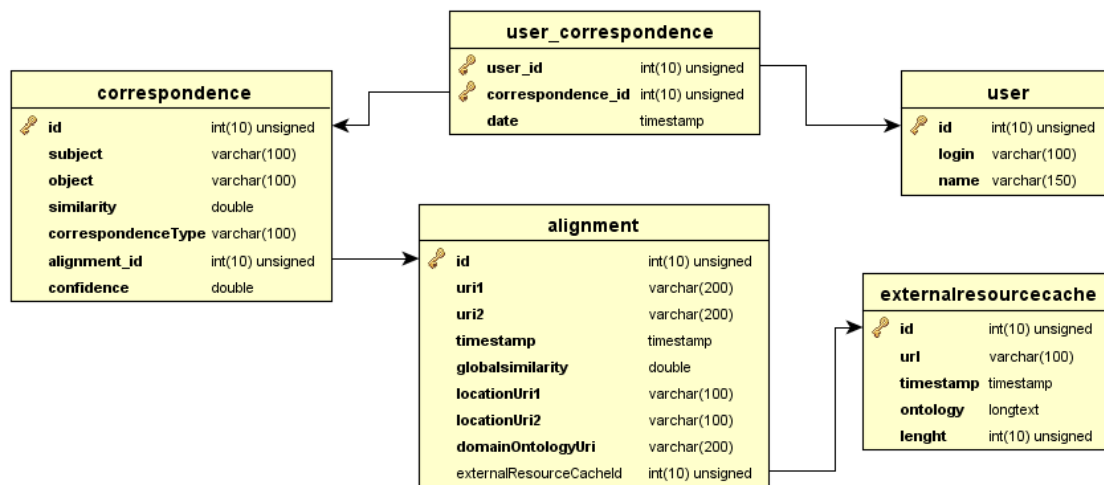


Figura 5.2 – Esquema relacional para o histórico de correspondências

A seguir temos uma descrição sobre cada tabela criada:

- *alignment*: guarda as informações sobre um processo de alinhamento executado. São armazenadas informações como as URIs das ontologias envolvidas no processo (*uri1*, *uri2* e *domainOntologyUri*), e a similaridade entre as duas ontologias (*globalsimilarity*). Informações adicionais são armazenadas como o local onde a ontologia está hospedada;
- *correspondence*: armazena informações sobre cada correspondência, ou seja, cada 5-tupla (sujeito, relacionamento, objeto, peso, grau de confiança). Caso tenha havido avaliação da correspondência, haverá uma entrada na tabela de relacionamento entre *user* e *correspondence*;
- *user*: esta tabela foi incluída na modelagem para armazenar qual usuário fez avaliação. A intenção é usá-la em um trabalho futuro junto com mais informações do usuário, constituindo um perfil que poderá ser usado para identificar avaliações de usuários que têm interesses em comum;
- *externalresourcecache*: tabela que guarda ontologias obtidas da Internet. Se for necessário o acesso a uma ontologia externa que já foi “baixada”, a versão aqui armazenada é utilizada. O campo *timestamp* é usado para ver se a versão local é antiga e deve-se baixar de novo a ontologia.

Para avaliar as funcionalidades incluídas na arquitetura do *SemMatcher*, alguns cenários de teste foram elaborados. Visando ilustrar os resultados obtidos para cada uma das funcionalidades, foi montado um cenário para comparar os resultados individualmente.

5.2 Experimentos

De forma a mostrar os resultados que conseguimos com o uso das técnicas propostas, escolhemos algumas das métricas sugeridas pela OAEI para avaliar ferramentas de *ontology matching* que foram explicadas em detalhes na Seção 2.5. Duas medidas foram escolhidas: precisão e cobertura. Para mensurar o desempenho, escolhemos o tempo de execução, levando em consideração uma máquina com a seguinte configuração: Processador core2duo, 3GB de RAM. Um comparativo com essas medidas será feito com relação à primeira versão do *matcher*. Nos nossos testes não fizemos uso das ontologias providas pela OAEI pelo fato de elas não conterem todos os tipos de relacionamentos considerados pelo *SemMatcher+* como, por exemplo, o *closeness* e *partOf*, o que geraria uma avaliação incompleta da ferramenta.

5.2.1 Uso de Sinônimos nas Regras Semânticas

A primeira funcionalidade incluída no *SemMatcher+* foi o uso do *WordNet* na execução das regras semânticas. Como foi descrito na Seção 4.1, durante a execução de cada regra, para cada conceito das ontologias comparadas, é procurado o equivalente na ontologia de domínio. A comparação é feita não apenas usando o rótulo do conceito, mas também com seus sinônimos de acordo com o *WordNet*.

O objetivo do experimento é avaliar se o uso de sinônimos pode melhorar o alinhamento final gerado. Executamos o *SemMatcher+* em um cenário referente ao domínio de Animais. As ontologias utilizadas estão disponíveis no Apêndice A. Na Figura 5.3 temos um comparativo da precisão e cobertura e no Quadro 5.1 uma comparação do tempo de execução e número de correspondências descobertas, ambos entre o *SemMatcher* e o *SemMatcher+* com uso do *WordNet*.

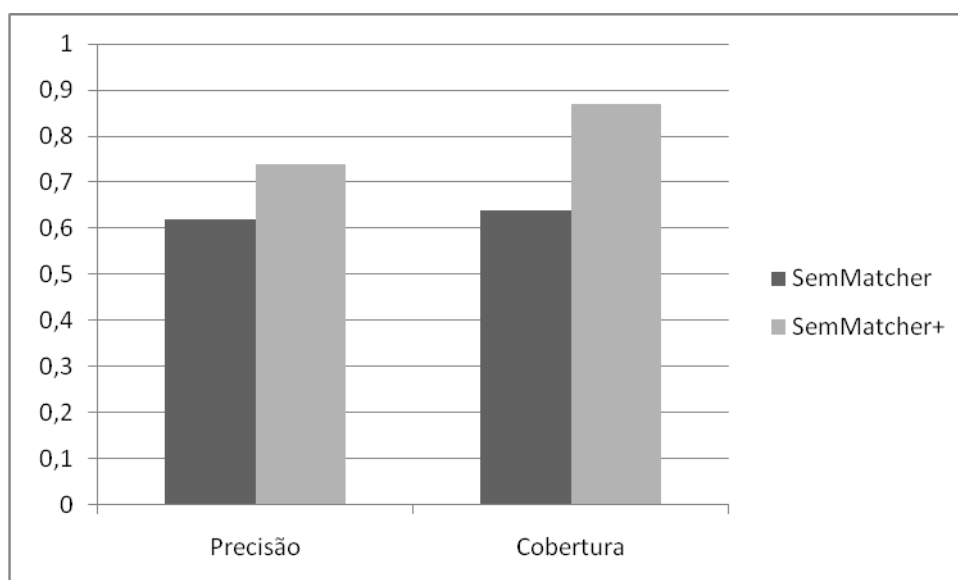


Figura 5.3 - Gráfico mostrando a evolução da precisão e cobertura nas versões do *SemMatcher* com e sem o uso do *WordNet*

	Tempo de execução	Número de correspondências encontradas
<i>SemMatcher</i>	14s	247
<i>SemMatcher+</i>	17s	301

Quadro 5.1 – Comparativo de tempo e número de correspondências com e sem o uso do *WordNet*

Podemos notar que houve uma melhora significativa para o cenário testado. Muitos conceitos que eram sinônimos não tinham suas correspondências descobertas, pois as comparações eram feitas apenas usando o rótulo exato do conceito contido na ontologia local. Notamos que mais de 50 novas correspondências foram encontradas como vemos no Quadro 5.1. O valor da cobertura aumentou, indicando que as novas correspondências encontradas são corretas, o que também é comprovado pelo fato de a precisão ter aumentado.

5.2.2 Conhecimento Externo

Outra funcionalidade do *SemMatcher+* é a busca por ontologias externas para descoberta de novo conhecimento que pode gerar novas correspondências no processo de *matching* corrente. Para isso, ontologias disponíveis na Web Semântica são usadas. Executamos as duas versões do *SemMatcher* em um cenário referente ao domínio de Educação. As ontologias utilizadas estão disponíveis no Apêndice B.

Ao buscar ontologias na Internet, um fator que impacta o desempenho do *SemMatcher+* diz respeito ao tempo de *download* da ontologia externa. Esse tempo extra torna a operação de *matching* mais demorada. Outro fato notado durante os experimentos é que algumas ontologias encontradas têm tamanho grande, chegando a quase 30MB e contendo um número muito grande de conceitos. Isso impacta não apenas no tempo de *download*, como também no tempo de carregamento da ontologia na memória. Por este motivo, foram feitos testes limitando o tamanho e, também, sem limitar o tamanho da ontologia externa.

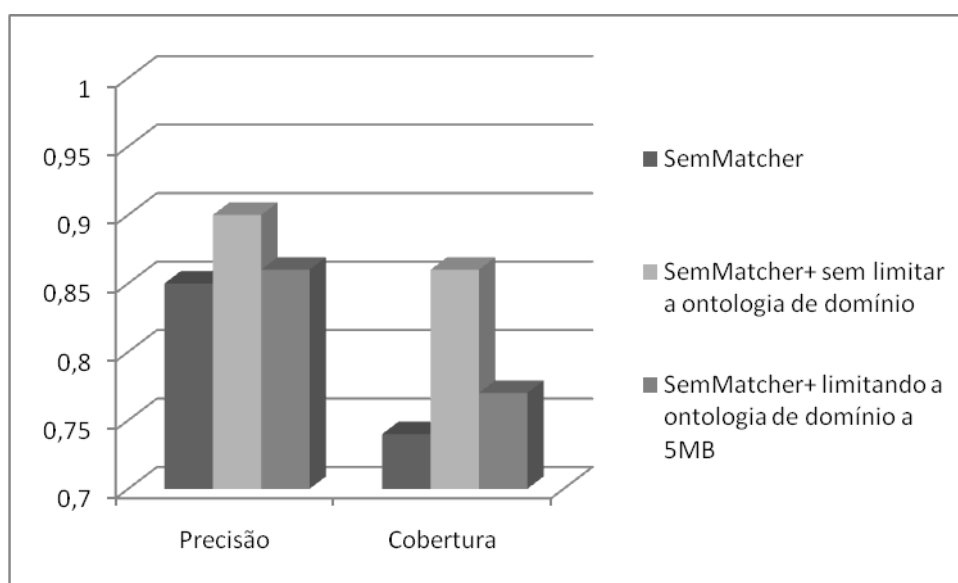


Figura 5.4 - Evolução da precisão e cobertura do *SemMatcher* para o *SemMatcher+* no domínio de educação com e sem o uso de conhecimento externo

	Tempo de execução	Número de correspondências encontradas
<i>SemMatcher</i>	10s	158
<i>SemMatcher+</i> sem limitar a ontologia externa	2m35s	173
<i>SemMatcher+</i> limitando a ontologia externa a 5MB	41s	162

Quadro 5.2 – Comparativo de tempo e número de correspondências do *SemMatcher* no domínio de educação com e sem o uso de conhecimento externo

Na Figura 5.4, observamos os valores de precisão e cobertura para cada uma das execuções do *SemMatcher* e *SemMatcher+*. No Quadro 5.2, o tempo de execução e o número de correspondências geradas pelo processo. Neste cenário também notamos que novas correspondências foram encontradas e que a precisão e, principalmente, a cobertura aumentaram, indicando que as novas correspondências são corretas. Ao trabalhar com uma ontologia de tamanho limitado, menos correspondências foram encontradas, mas ainda assim foram identificadas novas correspondências. Para ilustrar os resultados em um cenário diferente, construímos outras ontologias referentes ao domínio de Animais. As ontologias utilizadas são mostradas no Apêndice C.

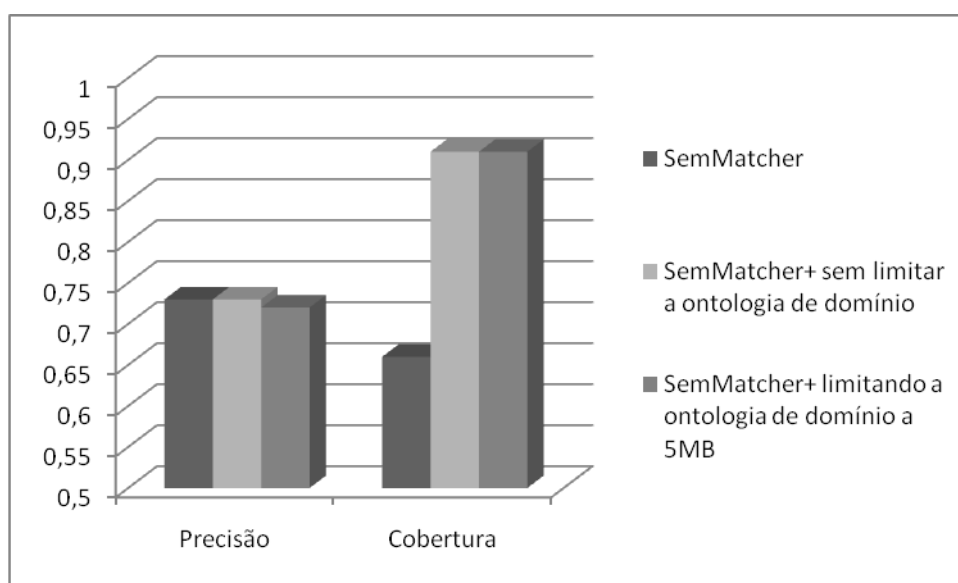


Figura 5.5 - Evolução da precisão e cobertura do *SemMatcher* no domínio de animais com e sem o uso de conhecimento externo

	Tempo de execução	Número de correspondências encontradas
<i>SemMatcher</i>	7s	174
<i>SemMatcher+</i> sem limitar a ontologia de domínio	36s	240
<i>SemMatcher+</i> limitando a ontologia de domínio a 5MB	30s	242

Quadro 5.3 – Comparativo de tempo e número de correspondências do *SemMatcher* no domínio de animais com e sem o uso de conhecimento externo

A Figura 5.5 ilustra a precisão e cobertura, e no Quadro 5.3 o tempo e o número de correspondências referentes a cada um dos testes executados. Apesar de a medida precisão continuar com o mesmo valor, a cobertura aumentou, indicando que muitas das correspondências novas que foram encontradas são corretas. Para refinar os resultados e eliminar essas correspondências geradas erroneamente, permitimos ao usuário avaliar as correspondências geradas. Os resultados dos experimentos com *feedback* do usuário são mostrados na próxima seção.

5.2.3 Feedback do Usuário

No *SemMatcher+* incluímos a possibilidade do usuário avaliar as correspondências geradas. Depois da execução do processo de *ontology matching*, o *SemMatcher+* exibe as correspondências finais geradas, cada uma com sua 5-tupla (sujeito, predicado, objeto, peso do relacionamento semântico, grau de confiança) correspondente. Neste ponto, as correspondências já estão armazenadas no banco de dados e são atualizadas quando o usuário as avalia. Na Figura 5.6, vemos os dados das correspondências geradas, além de um último campo (“Reject”) para que o usuário possa avaliar as correspondências.

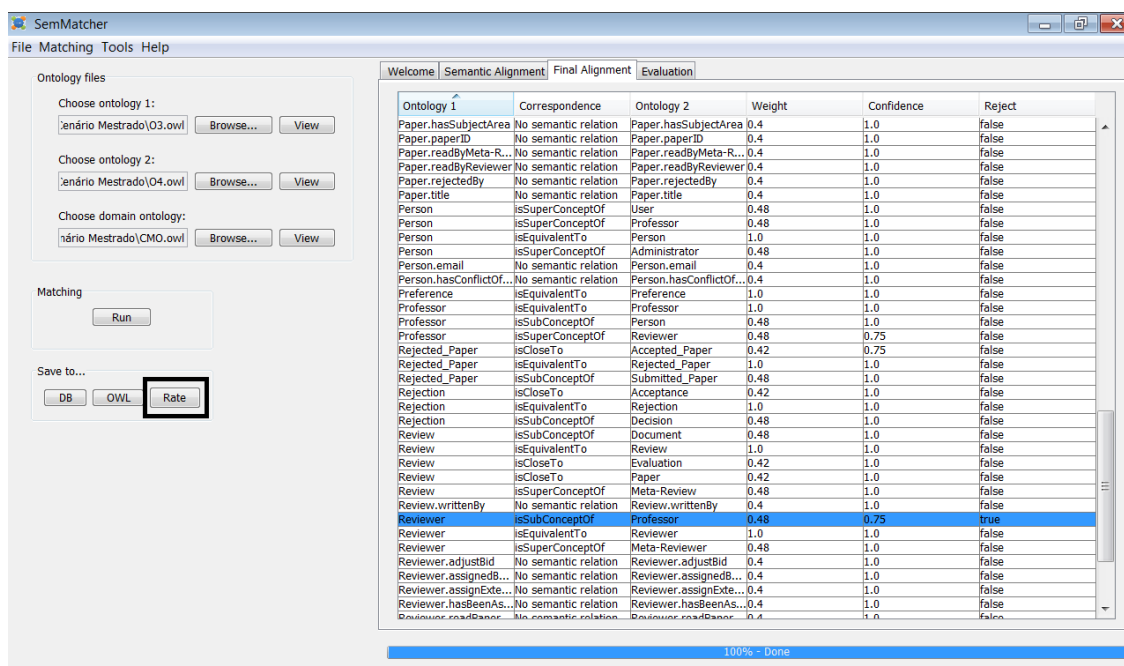


Figura 5.6 – Tela com resultado final para avaliação do usuário.

Ao clicar no campo *Reject* da correspondência desejada, ela é marcada para ser rejeitada para aquela execução. Ao clicar no botão “Rate” (destacado à esquerda da figura), as correspondências são atualizadas no banco de dados. Como explicado na Seção 4.5, ao fazer isso, as correspondências marcadas têm seu grau de confiança

atualizados para 0.0 na execução atual. Quando outra execução for feita com ontologias parecidas, é calculada a média dos graus de confiança de execuções anteriores, e uma média ponderada é calculada para encontrar o valor do grau de confiança para a correspondência atual. Todo este processo foi explicado em detalhes no capítulo anterior, mais precisamente na Seção 4.1.4.

Em seu uso no sistema SPEED [Paz 2010], a rejeição de correspondências é feita de forma indireta na tela de apresentação dos resultados da consulta. Quando uma consulta é feita, os alinhamentos são usados pelo SPEED para reformular a consulta para outro *peer* e os seus resultados são integrados aos resultados locais. Na tela mostrada na Figura 5.7, o usuário vê o resultado da sua consulta e, se algum resultado foi considerado incorreto, ele pode informar que aquele resultado é inválido. Ao invalidar um resultado, são identificadas as correspondências que o gerou, e as mesmas são avaliadas de forma negativa (tornando o seu grau de confiança zero).

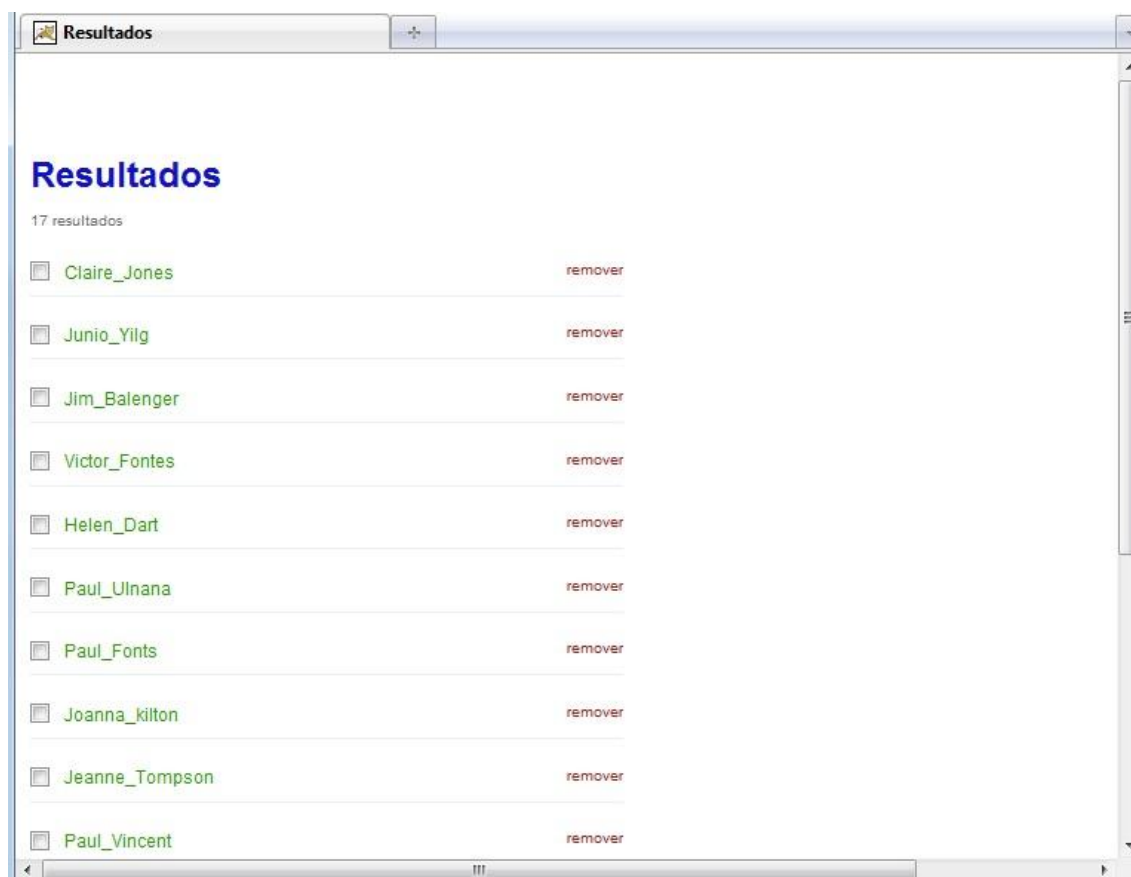


Figura 5.7 - Tela de avaliação de resultados de consultas no sistema SPEED [Paz 2010]

Como descrito no capítulo anterior, quando atualizamos o grau de confiança levando em consideração o histórico de correspondências, fazemos uma média ponderada entre o grau de confiança atual e a média dos graus de confiança anteriores.

Nos experimentos feitos, usamos os pesos de 0.6 para o valor do grau de confiança atual e 0.4 para a média dos graus de confiança anteriores. O peso maior para o grau de confiança atual é justificado pelo fato de ele ter sido encontrado no processo atual, por meio de uma ontologia de domínio informada pelo usuário, ou usando a ontologia externa encontrada na Internet que possui conceitos das ontologias do processo corrente.

Outro fator para o uso de um peso maior para o grau de confiança atual é o fato de que, se o peso da média dos graus de confiança anteriores for grande, uma única rejeição pode tornar o valor do grau de confiança atual muito baixo (se houver apenas uma correspondência anterior e ela tiver valor zero para o grau de confiança). Feito o cálculo, estamos eliminando as correspondências que têm grau de confiança menor que 0.7.

Além do parâmetro para o cálculo do novo grau de confiança, configuramos o limiar usado na busca de alinhamentos passados para calcular a média descrita no parágrafo anterior. Escolhemos filtrar alinhamentos com similaridade maior que 0.7 entre suas ontologias e as ontologias do processo atual. Um limiar menor que esse indicaria que a ontologia tem conceitos realmente diferentes das ontologias atuais e, portanto, deve tratar de um domínio diferente.

Usamos um conjunto de 12 ontologias referentes ao domínio de Educação (disponíveis no Apêndice D) para exemplificar uma execução do *SemMatcher+* e o uso de histórico de correspondências. Para mostrar como acontece a diminuição do grau de confiança para cada execução, escolhemos uma correspondência considerada inválida e mostramos o quanto seu grau de confiança vai sendo modificado a cada execução e a cada avaliação do usuário. Chamaremos cada uma das 12 ontologias de O_n , onde n é um número de 1 a 12 indicando qual ontologia está sendo usada.

Escolhemos a correspondência (*Accepted_Paper*, *Rejected_Paper*, *isCloseTo*, 0.42, 1.0) para mostrar como exemplo, pois sabemos que no mundo real esses dois conceitos são disjuntos. Um *Paper* não pode ser aceito e rejeitado ao mesmo tempo. O Quadro 5.4 mostra como o grau de confiança se modifica a cada execução usando o cálculo explicado na Seção 4.4.

Execução	Ontologias	Média dos graus de confiança anteriores	Grau de confiança atual	Novo grau de confiança	Avaliação do usuário	Valor final
1	O ₁ e O ₂	-	1	1	Não rejeitado	1
2	O ₃ e O ₄	1	1	$1 \times 0.4 + 1 \times 0.6 = 1$	Rejeitado	0
3	O ₅ e O ₆	0.5	1	$0.5 \times 0.4 + 1 \times 0.6 = 0.8$	Rejeitado	0
4	O ₇ e O ₈	0.33	1	$0.33 \times 0.4 + 1 \times 0.6 = 0.73$	Rejeitado	0
5	O ₉ e O ₁₀	0.25	1	$0.25 \times 0.4 + 1 \times 0.6 = 0.7$	Rejeitado	0
6	O ₁₁ e O ₁₂	0.2	1	$0.2 \times 0.4 + 1 \times 0.6 = 0.68$	(Correspondência rejeitada por ser inferior ao <i>threshold</i>)	0

Quadro 5.4 - Atualização do grau de confiança durante as execuções

Na primeira execução o grau de confiança encontrado é 1.0 pois o alinhamento foi encontrado a partir da ontologia de domínio fornecida como parâmetro. Nesta primeira execução não fizemos uma avaliação, e o valor do grau de confiança se manteve em 1.0. A cada execução seguinte, fomos rejeitando a correspondência e, cada vez que a média ponderada foi calculada, o valor do grau de confiança foi diminuindo. Na sexta execução, ao final do cálculo da média ponderada, o novo grau de confiança é menor que o limiar 0.7 dito anteriormente, então a correspondência é automaticamente descartada atualizando o seu grau de confiança para 0.0.

Neste teste, além da correspondência em questão, outras quatro foram eliminadas durante o processo por também serem inválidas. Vemos assim que o *feedback* do usuário é usado como um ajuste fino do alinhamento final, retirando correspondências incorretas identificadas pelo usuário, que geralmente é o maior especialista sobre o domínio.

O tempo médio de cada execução indicada no Quadro 5.4 foi aproximadamente 10s (incluindo o tempo de busca a ontologias externas, mas sem contar o tempo de *download* nem o tempo de avaliação do usuário). A eliminação das correspondências não causou uma melhoria notável na precisão e cobertura porque o número de correspondências total é grande em relação ao número de correspondências inválidas eliminadas, mas correspondências que realmente eram inválidas foram eliminadas, levando a um melhor resultado.

5.3 Considerações Finais

Implementar as novas funcionalidades propostas no *SemMatcher+* possibilitou a realização dos experimentos considerando a versão anterior e a versão atual. Em nossos experimentos, vimos que o uso de fontes auxiliares acrescenta novo conhecimento que muitas vezes não é encontrado apenas analisando os dados locais.

Quando nos referimos a ontologias disponíveis na Internet, é grande a quantidade de conhecimento disponível. O acesso a esse conhecimento não é fácil sem um meio de fazer buscas. Os sites *Swoogle* e o *Watson* fornecem uma interface de busca por palavra-chave para a consulta a ontologias disponíveis na Internet. Buscar ontologias usando como palavras-chave todos os conceitos presentes na ontologia é um critério de busca bastante restritivo e que costuma não retornar resultados. Como nosso objetivo é buscar conhecimento adicional, focamos na busca por ontologias que contenham conceitos para os quais ainda não foram encontradas correspondências.

Nossos experimentos mostraram que, apesar da piora no desempenho devido aos tempos de acesso e *download* das ontologias, conseguimos encontrar novas correspondências relevantes e corretas. A melhora do alinhamento foi comprovada com o aumento das medidas precisão e cobertura.

Mesmo encontrando muitas correspondências relevantes, também podem aparecer correspondências inválidas. Para permitir ao usuário refinar o alinhamento gerado, oferecemos ao mesmo a possibilidade de rejeitar correspondências incorretas. O *SemMatcher+* atualiza as correspondências rejeitadas no banco, modificando o seu grau de confiança. Nossos experimentos mostraram como rejeições seguidas a uma correspondência a tornam inválida. No próximo capítulo, veremos as conclusões e trabalhos futuros a serem realizados.

6. Conclusão e Trabalhos Futuros

O processo de encontrar correspondências entre ontologias é útil para diversos campos de pesquisa, como mostramos anteriormente. Para solucionar o problema da heterogeneidade entre ontologias, muitas técnicas de *ontology matching* foram desenvolvidas. As primeiras técnicas eram baseadas apenas na sintaxe e na estrutura da ontologia e falhavam por não levarem a semântica em consideração. No aspecto semântico, o uso de ontologias de referência para descoberta de relacionamentos é uma boa abordagem, porém nem sempre a ontologia de referência fornecida como entrada contém todo o conhecimento necessário para a descoberta de correspondências.

A Web Semântica vem para aumentar a semântica na Internet, possibilitando buscas mais elaboradas, combinação de informações para inferir novo conhecimento, dentre outras facilidades. Com isso, o número de ontologias disponíveis na Internet tende a crescer cada vez mais. Alguns *crawlers* já existem especificamente para tratar de ontologias, permitindo buscas que consideram não apenas a sintaxe, como também a semântica. Sistemas como esses podem ser usados para obter novas ontologias para serem usadas como ontologia de referência e gerar novas correspondências.

O *feedback* do usuário também é uma importante fonte de conhecimento a ser explorada, pois o usuário é o real especialista no domínio, e sua avaliação sobre o alinhamento gerado ajuda a eliminar correspondências incorretas no resultado. Nas próximas seções veremos a contribuição deste trabalho e os trabalhos futuros pensados.

6.1 Contribuições

A contribuição deste trabalho pode ser descrita em quatro partes:

- *Descoberta de novo conhecimento usando ontologias encontradas na Internet.*

A abordagem proposta para descoberta de novo conhecimento faz uso de sites que indexam e raciocinam sobre ontologias disponíveis na Internet, permitindo buscas a novas ontologias que servem como uma fonte de conhecimento adicional. Notamos que apenas a ontologia de domínio fornecida como entrada muitas vezes não é suficiente para descobrir todos os relacionamentos existentes entre os conceitos das ontologias locais. Os experimentos confirmaram que novas correspondências podem ser encontradas a partir de ontologias obtidas a partir de engenhos de busca como o *Swoogle* e *Watson*.

- *Uso de dicionário de sinônimos na descoberta de relacionamentos na ontologia de domínio*

Quando identificamos conceitos semelhantes entre as ontologias locais e as de domínio para inferência de novos relacionamentos, a comparação feita no *SemMatcher* considerava apenas conceitos com mesmo nome em grafia. Essa abordagem não permite encontrar muitas correspondências por eliminar conceitos que eram sinônimos, mas tinham grafia diferente. O uso do *WordNet* como dicionário de sinônimos para a comparação de conceitos entre as ontologias locais e a de domínio fez com que mais correspondências fossem encontradas e, conseqüentemente, aumentou o potencial semântico do processo.

- *Reformulação das regras para descoberta de correspondências entre propriedades*

A maioria das ferramentas de *ontology matching* foca na descoberta de correspondências entre conceitos, porém a descoberta de correspondências entre propriedades também é muito importante para vários sistemas que fazem uso dessas ferramentas, como os sistemas PDMS que precisam reformular consultas. As correspondências do tipo equivalência, subpropriedade e superpropriedade foram refinadas para levar em consideração não apenas informações sobre a propriedade, mas também incluir o conceito que a contém na regra.

- *Possibilidade de refinamento do conjunto de correspondências encontradas*

Para eliminar correspondências erroneamente encontradas, o nosso processo permite a rejeição de correspondências inválidas. Essas avaliações são armazenadas e usadas em operações posteriores visando à descoberta de correspondências. Correspondências avaliadas de forma negativa repetidas vezes são eliminadas em alinhamentos futuros que envolvam ontologias do mesmo domínio.

6.2 Trabalhos Futuros

Alguns pontos deste trabalho podem ser melhorados, e outras funcionalidades foram deixadas para um futuro trabalho, dentre elas destacamos:

- *Uso de ontologias externas também para validação de correspondências já existentes*

Atualmente, quando buscamos ontologias, usamos conceitos que ainda não têm correspondência. Porém, a ontologia externa poderia ser usada também como uma forma de validar as correspondências existentes. Ao fazer a busca no *Swoogle* e *Watson*, é necessário fornecer palavras-chave para o engenho de busca. Ao usar

muitas palavras, o critério se torna muito restritivo e acaba por não retornar resultados. Quando focamos nos conceitos que ainda não possuem correspondência, diminuimos o número de conceitos possíveis a serem usados na busca, porém, poderíamos usar os demais conceitos para encontrar ontologias que validassem as correspondências já existentes.

- *Buscar formas de melhorar o desempenho no carregamento de ontologias muito grandes*

Ao buscar ontologias na Internet, às vezes são retornadas ontologias muito grandes. O carregamento desses arquivos é lento, problema esse que não acontece apenas na nossa ferramenta, como também em outras ferramentas testadas. A busca por uma API de manipulação de ontologias mais eficiente ou a implementação de uma, pode ser considerado um trabalho a ser realizado no futuro.

- *Explorar mais de uma ontologia externa*

Atualmente usamos apenas uma ontologia externa para descoberta de conhecimento adicional na Internet. Devido a problemas de desempenho e largura de banda, explorar várias ontologias poderia ter um impacto muito negativo no tempo total do processo. Sistemas como o SCARLET fazem acesso a várias ontologias, mas armazenam essas ontologias em *cache* e possuem estratégias de indexação e armazenamento que auxiliam o acesso ao conhecimento armazenado nas ontologias. Atualmente, armazenamos as ontologias obtidas da Internet, mas o tempo de carregamento para ontologias grandes é um fator determinante para o aumento no tempo de execução. O estudo de novas técnicas de armazenamento que possibilitem acesso rápido ao conhecimento nas ontologias armazenadas é um desafio que pode ser melhorado em futuras versões da ferramenta.

- *Explorar conhecimento existente nas próprias ontologias locais na descoberta de relacionamentos*

A própria ontologia local às vezes possui conhecimento útil a ser utilizado para a descoberta de correspondências. Atualmente apenas as ontologias de domínio e a ontologia externa são usadas para descobrir novos conhecimentos, mas as próprias ontologias locais possuem relacionamentos que podem ser considerados na descoberta de correspondências. Supondo que temos as ontologias O_1 e O_2 sendo comparadas. Na ontologia O_1 temos que *Artigo* é subconceito de *Publicação* e não há informação referente a *Livro*. Se na ontologia O_2 temos que *Livro* é subconceito de *Publicação*, podemos inferir a partir do conhecimento que *Livro* e *Artigo* têm

Publicação como superconceito, que *Livro* e *Artigo* são conceitos próximos (*IsCloseTo*).

Referências

- [d'Aquin et al. 2007] M. d'Aquin, C. Baldassarre, L. Gridinoc, M. Sabou, S. Angeletou, E. Motta, "WATSON: Supporting next generation of semantic web applications". In Proceedings of the WWW/Internet Conference, pages 363-371, Vila Real, Portugal, 2007.
- [Aumuellner et al. 2005] D. Aumuellner, H. Do, S. Massmann, E. Rahm, "Schema and ontology matching with COMA++". In Proceedings of SIGMOD, pages 906-908, Baltimore, USA, 2005.
- [Belhajjame et al. 2010] K. Belhajjame, N. W. Paton, S. M. Embury, A. A. A. Fernandes, C. Hedeler, "Feedback-Based Annotation, Selection and Refinement of Schema Mappings for Dataspaces". In Proceedings of EDBT, pages 573-584, Lausanne, Switzerland, 2010.
- [Berners-Lee et al. 2001] T. Berners-Lee, J. Hendler, O. Lassila, "The Semantic Web". Scientific American no. 284, pages 34-43, 2001.
- [Bouquet et al. 2003] P. Bouquet, L. Serafini, S. Zanobini, "Semantic coordination: a new approach and an application". In Proceedings of the International Semantic Web Conference (ISWC), pages 130-145, Florida, USA, 2003.
- [Bouquet et al. 2005] P. Bouquet, J. Euzenat, E. Franconi, L. Serafini, G. Stamou, S. Tessaris, "Specification of a common framework for characterizing alignment". KnowledgeWeb Project Deliverable D2.2.1, 2005.
- [Bouquet et al. 2006] P. Bouquet, L. Serafini, S. Zanobini, S. Sceffer, "Bootstrapping semantics on the web: meaning elicitation from schemas". In Proceedings of International World Wide Web Conference (WWW), pages 235-251, Southampton, United Kingdom, 2006.
- [Castano et al. 1998] S. Castano, V. Antonellis, M. G. Fugini, B. Pernici, "Conceptual Schema Analysis: Techniques and Applications". In ACM Transactions on Database Systems, Volume 23, N° 3, pages 286-333, 1998.
- [Castano et al. 2003] S. Castano, A. Ferrara, S. Montanelli, "H-match: an Algorithm for Dynamically Matching Ontologies in Peer-based Systems Peer-based Systems". In

Proceedings of the 1st VLDB Int. Workshop on Semantic Web and Databases (SWDB), pages 231-250, Berlin, Germany, 2003.

[DAML 2011] “DAML.org”. Accessed on 13/04/2011 at <<http://www.daml.org/>>

[Duan et al. 2010] S. Duan, A. Fokoue, K. Srinivas, “One size does not fit all: Customizing Ontology Alignment Using User Feedback”. In Proceedings of International Semantic Web Conference (ISWC), pages 177-192, Shanghai, China, 2010.

[Dzbor et al. 2004] M. Dzbor, E. Motta, J. Domingue, "Opening up Magpie via semantic services". In Proceedings of International Semantic Web Conference (ISWC), pages 635-649, Hiroshima, Japan, 2004.

[Euzenat 2004] J. Euzenat, "An API for ontology alignment". In Proceedings of the International Semantic Web Conference (ISWC), pages 698-712, Hiroshima, Japan, 2004.

[Euzenat *et al.* 2005] J. Euzenat, M. Ehrig, R. G. Castro,. "Towards a methodology for evaluating alignment and matching algorithms". Technical Report, Ontology Alignment Evaluation Initiative (OAEI), 2005.

[Euzenat and Shvaiko 2007a] J. Euzenat, P. Shvaiko, “Ontology Matching”. Springer-Verlag, Heidelberg, 2007.

[Euzenat and Shvaiko 2007b] J. Euzenat, P. Shvaiko, “Semantic precision and recall for ontology alignment evaluation”. In Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pages 348-353, Hyderabad, India, 2007.

[Fagin et al. 2009] R. Fagin, L. M. Haas, M. A. Hernández, R. J. Miller, L. Popa, and Y. Velegrakis. "Clio: Schema mapping creation and data exchange". In "Conceptual Modeling: Foundations and Applications", pages 198-236, Springer, Heidelberg, 2009.

[Fellbaum 1998] C. Fellbaum, "WordNet: An Electronic Lexical Database". Cambridge, MA: MIT Press, 1998.

[Fensel 2004] D. Fensel, "Ontologies: a silver bullet for knowledge management and electronic commerce". Springer, Heidelberg, 2nd Edition, 2004.

[Finin 2005] T. W. Finin, L. Ding, R. Pan, A. Joshi, P. Kolari, A. Java, Y. Peng, "Swoogle: Searching for knowledge on the semantic web". In Proceedings of the

National Conference on Artificial Intelligence, pages 1682-1683, Pittsburgh, Pennsylvania, 2005.

[Garey and Johnson 1979] M. Garey, D. Johnson. "Computers and intractability: a guide to the theory of NP-completeness". W. H. Freeman & Co., New York, 1979.

[Giunchiglia et al. 2009] F. Giunchiglia, P. Shvaiko, M. Yatskevich. "Semantic matching with S-Match". In "Semantic Web Information Management", Springer, 2009.

[Gruber 1993] T. R. Gruber. "A Translation Approach to Portable Ontology Specifications". In Knowledge Acquisition, Volume 5, Issue 2, pages 199-220, 1993.

[Guarino 1998] N. Guarino, "Formal Ontology and Information Systems". In Proceedings of International Conference on Formal Ontologies in Information Systems, Trento, Italy, 1998.

[Halevy et al. 2005] A. Halevy, N. Ashish, D. Bitton, M. Carey, D. Draper, J. Pollock, A. Rosenthal, V. Sikka, "Enterprise information integration: successes, challenges and controversies". In Proceedings of International Conference on Management of Data (SIGMOD), pages 778-787, Baltimore, USA, 2005.

[JAWS 2011] "Java API for WordNet Searching (JAWS)". Accessed on 13/04/2011 at <<http://lyle.smu.edu/~tspell/jaws/index.html>>

[Jean-Mary et al. 2010] Y. R. Jean-Mary, E. P. Shironoshita, M. R. Kabuka, "ASMOV: Results for OAEI 2010", In the OAEI 2010 Campaign. International Workshop on Ontology Matching, Shanghai, China, 2010.

[JWI 2011] "MIT Java Wordnet Interface (JWI)". Accessed on 13/04/2011 at <<http://projects.csail.mit.edu/jwi/>>

[JWNL 2011] "SourceForge.net: jwordnet". Accessed on 13/04/2011 at <<http://jwordnet.sourceforge.net/>>

[Quix et al. 2010] C. Quix, M. Pascan, P. Roy, D. Kensche, "Semantic Matching of Ontologies", In Proceedings of International Workshop on Ontology Matching, pages 238-240, Shanghai, China, 2010.

[Lucene 2011] "Apache Lucene - Overview". Accessed on 13/04/2011 at <<http://lucene.apache.org/java/docs/index.html>>

- [Madhavan et al. 2001] J. Madhavan, P. Bernstein, E. Rahm, "Generic schema matching with Cupid", In Proceedings of International Conference on Very Large DataBases (VLDB), pages 49-58, Roma, Italy, 2001.
- [NeOn 2011] "Welcome to the NeOn Project". Accessed on 13/04/2011 <<http://www.neon-project.org/web-content/>>
- [OAEI 2011] "Ontology Alignment Evaluation Initiative". Accessed on 13/04/2011 at <<http://oaei.ontologymatching.org/>>
- [OWL 2011] "OWL Web Ontology Language Reference". Accessed on 13/04/2011 at <<http://www.w3.org/TR/owl-ref/>>
- [Paulheim et al. 2007] H. Paulheim, M. Rebstock, J. Fengel, "Context-sensitive referencing for ontology mapping disambiguation". In Proceedings of Workshop on Context and Ontologies Representation and Reasoning, Computer Science Research Report #115, pages 47-56, Roskilde University, 2007.
- [Paz 2010] T. H. F. Paz, "Considerando o feedback do usuário para melhoria dos resultados de consultas.", Federal University of Pernambuco (UFPE/CIn). Undergraduate Conclusion Monograph, Recife, Brazil, 2010.
- [Pereira 2008] T. P. A. Pereira, "Mapeamento Semântico de Ontologias no SPEED". Federal University of Pernambuco (UFPE/CIn). Undergraduate Conclusion Monograph, Recife, Brazil, 2008.
- [Po and Bergamaschi 2010] L. Po, S. Bergamaschi, "Automatic lexical annotation applied to the SCARLET ontology matcher". In Proceedings of Asian Conference on Intelligent Information and Database Systems (ACIIDS), pages 144-153, Hue City, Vietnam, 2010.
- [Pinto et al. 1999] H. S. Pinto, A. Gomez-Perez, J. P. Martins, "Some issues on Ontology Integration". In Proceedings of Workshop on Ontologies and Problem Solving Methods: Lessons Learned and Future Trends on International Joint Conference on Artificial Intelligence (IJCAI), Volume 18, Issue 7, pages 1-12, Stockholm, Sweden, 1999.
- [Pires et al. 2006] C. E. S. Pires, B. F. Lóscio, A. C. Salgado, "Gerenciamento de Dados em Sistemas P2P". In Proceedings of Brazilian Symposium on Databases (SBBD), page 310, Florianópolis, Brazil, 2006.

[Pires et al. 2009] C. E. S. Pires, D. Souza, T. Pachêco, A. C. Salgado, "SemMatcher: A Tool for Matching Ontology-based Schemas". In Brazilian Symposium on Data Bases (SBBD'09), Fortaleza, Brazil, 2009.

[Rahm and Bernstein 2001] E. Rahm, P. A. Bernstein, "A survey of approaches to automatic schema matching", In The VLDB Journal (International Journal on Very Large Data Bases), Volume 10, nº 4, pages 334-350, Roma, Italy, 2001.

[RDF 2011] "Resource Description Framework (RDF)". Accessed on 13/04/2011 at <<http://www.w3.org/RDF/>>

[Rijsbergen 1979] C. J. Rijsbergen, "Information Retrieval", 2nd Edition, Stoneham, MA: Butterworths, 1979.

[Sabou et al. 2007] M. Sabou, J. Gracia, S. Angeletou, M. d'Aquin, E. Motta, "Evaluating the semantic web: A task-based approach." In K. Aberer, K.-S. Choi, N. F. Noy, D. Allemang, K.-I. Lee, L. J. B. Nixon, J. Golbeck, P. Mika, D. Maynard, R. Mizoguchi, G. Schreiber, and P. Cudré-Mauroux, editors, ISWC/ASWC, volume 4825 of Lecture Notes in Computer Science, pages 423–437. Springer, 2007.

[Sabou et al. 2008] M. Sabou, M. d'Aquin, E. Motta, "Exploring the semantic web as background knowledge for ontology matching", In Journal of Data Semantics, Volume 11, pages 156-190, 2008.

[Sah et al. 2008] M. Sah, W. Hall, D. C. De Roure, "SemWeB: A Semantic Web Browser for Supporting the Browsing of Users using Semantic and Adaptive Links", In Proceedings of International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems Doctoral Consortium, pages 431-436, Hannover, Germany, 2008.

[Shvaiko and Euzenat 2008] P. Shvaiko, J. Euzenat, "Ten challenges for ontology matching". In On the move to Meaningful Internet Systems (OTM), pages 1164-1182, Monterrey, Mexico, 2008.

[Souza 2009] D. Souza, "Using Semantics to Enhance Query Reformulation in Dynamic Distributed Environments", Ph.D. Thesis, Federal University of Pernambuco (UFPE/CIn). Recife, Brazil, 2009.

[Sung et al. 2006] L. G. A. Sung, N. Ahmed, R. Blanco, H. Li, M. A. Soliman, D. Hadaller, "A Survey of Data Management in Peer-to-peer Systems". In School of Computer Science, Tech. Report CS-2006-18, University of Waterloo, pages 1-51, Canada, 2006.

[Swoogle 2011] “Swoogle”. Accessed on 13/04/2011 at <<http://www.swoogle.com>>

[Tierney and Jackson 2008] B. Tierney, M. Jackson, "C-SAW - Contextual semantic alignment of ontologies: using negative semantic reinforcement", In Proceedings of ACM Symposium on Applied Computing, pages 2346-2347, Fortaleza, Brazil, 2008.

[Xiao 2006] H. Xiao, “Query Processing for Heterogeneous Data Integration using Ontologies”. PhD Thesis. University of Illinois at Chicago, Chicago, USA, 2006.

[W3C 2011] “World Wide Web Consortium (W3C)”. Accessed on 13/04/2011 at <<http://www.w3.org/>>

Apêndice A

Neste apêndice, temos as ontologias do domínio de animais usadas no experimento do *WordNet*. As ontologias comparadas são apresentadas a seguir (a ontologia de domínio foi dividida em partes devido ao seu tamanho):

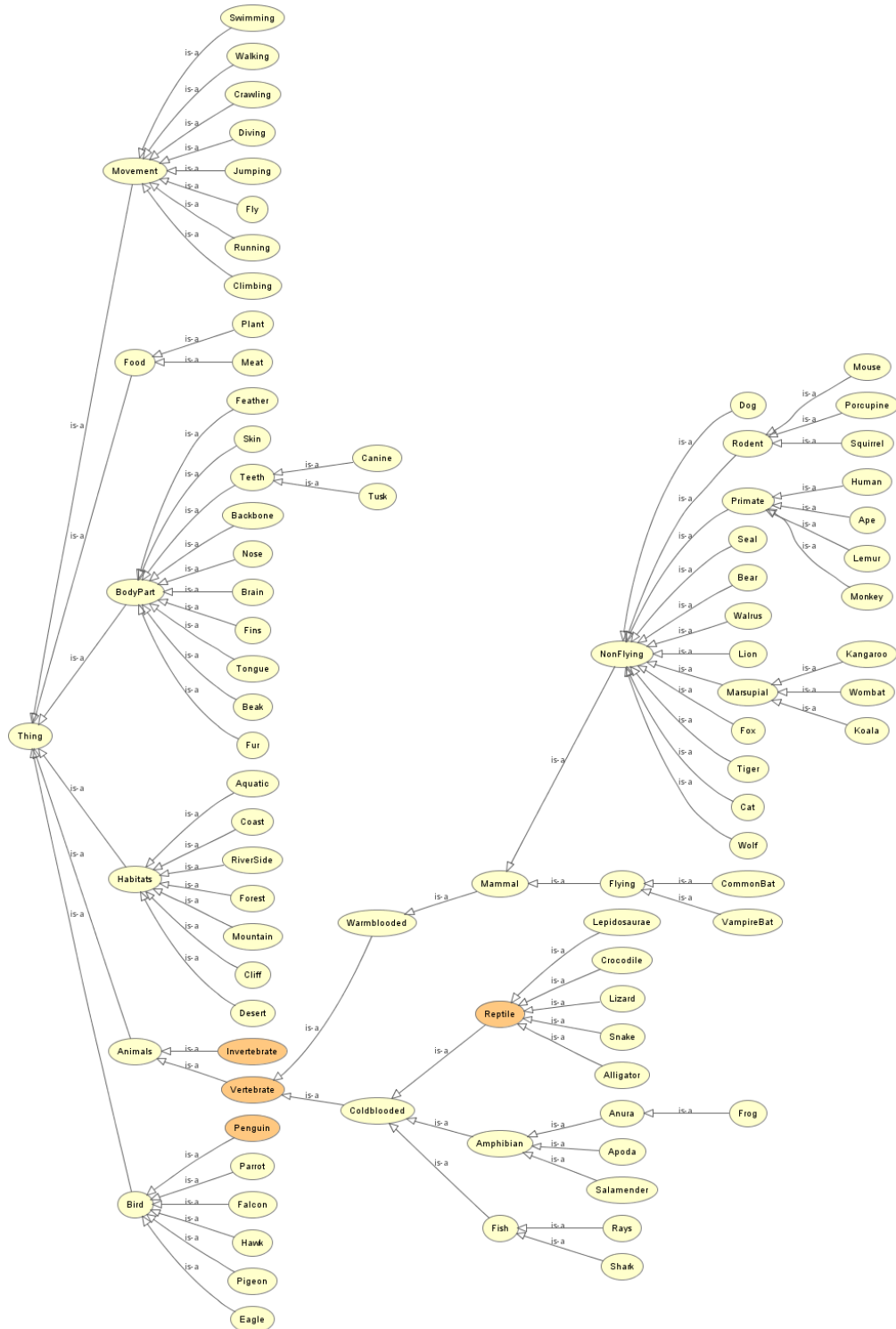


Figura A.1 - Ontologia de entrada 1 do domínio de animais do experimento de uso *WordNet*

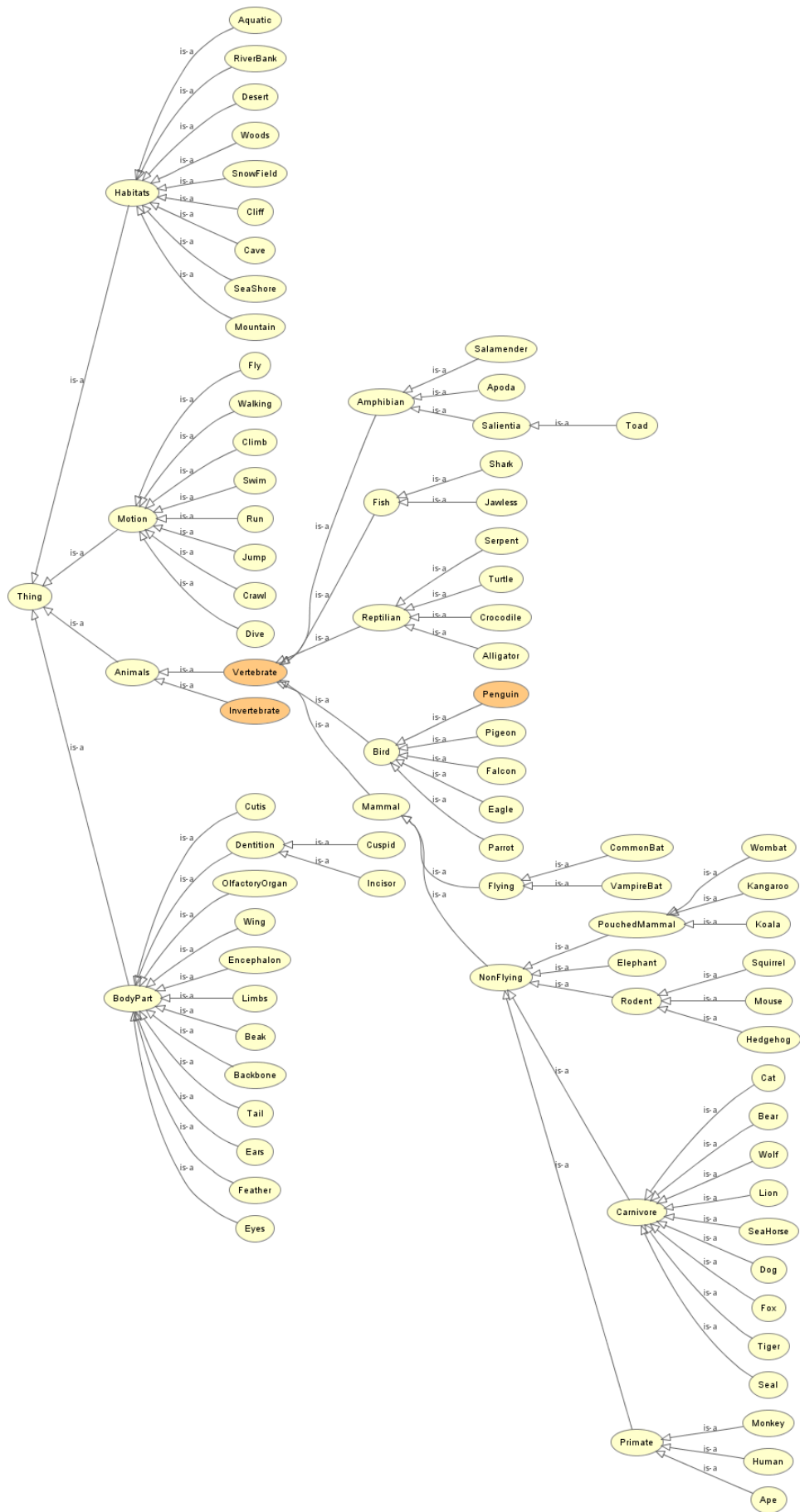


Figura A.2 - Ontologia de entrada 2 do domínio de animais do experimento de uso *WordNet*

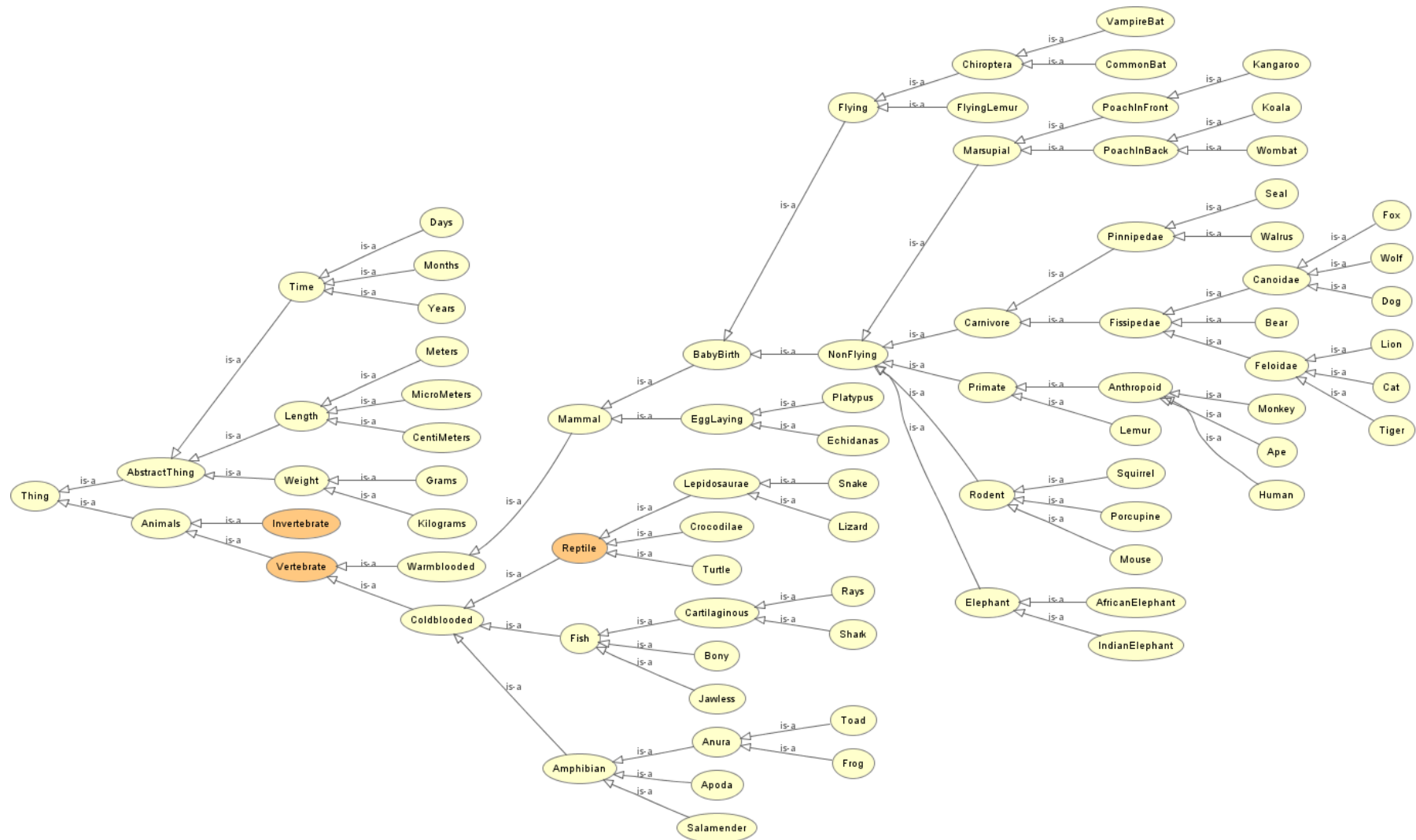


Figura A.3 - Primeira parte da ontologia de domínio usada no experimento de uso *WordNet*



Figura A.4 - Segunda parte da ontologia de domínio usada no experimento de uso *WordNet*



Figura A.5 - Terceira parte da ontologia de domínio usada no experimento de uso *WordNet*

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns="http://knowledgeweb.semanticweb.org/heterogeneity/alignment"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#">
  <Alignment>
    <xml>yes</xml>
    <level>0</level>
    <type>**</type>
    <onto1>http://a.com/ontology#</onto1>
    <onto2>http://a.com/ontology#</onto2>
    <uri1>http://a.com/ontology#</uri1>
    <uri2>http://a.com/ontology#</uri2>
    <map>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Snake"/>
        <entity2 rdf:resource="http://a.com/ontology#Serpent"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Snake"/>
        <entity2 rdf:resource="http://a.com/ontology#Alligator"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Snake"/>
        <entity2 rdf:resource="http://a.com/ontology#Crocodile"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#CommonBat"/>
        <entity2 rdf:resource="http://a.com/ontology#CommonBat"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#CommonBat"/>
        <entity2 rdf:resource="http://a.com/ontology#Flying"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#CommonBat"/>
        <entity2 rdf:resource="http://a.com/ontology#VampireBat"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Bird"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Vertebrate"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Pigeon"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Parrot"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Eagle"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Falcon"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Bird"/>
        <entity2 rdf:resource="http://a.com/ontology#Penguin"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Jumping"/>
        <entity2 rdf:resource="http://a.com/ontology#Jump"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Jumping"/>
        <entity2 rdf:resource="http://a.com/ontology#Motion"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Jumping"/>
        <entity2 rdf:resource="http://a.com/ontology#Swim"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Jumping"/>
        <entity2 rdf:resource="http://a.com/ontology#Fly"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://a.com/ontology#Jumping"/>
        <entity2 rdf:resource="http://a.com/ontology#Climb"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
    </map>
  </Alignment>
</rdf:RDF>

```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

<Cell>	<entity1 rdf:resource="http://a.com/ontology#Habitats"/> <entity2 rdf:resource="http://a.com/ontology#SeaShore"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Habitats"/> <entity2 rdf:resource="http://a.com/ontology#Mountain"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Habitats"/> <entity2 rdf:resource="http://a.com/ontology#Cave"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Habitats"/> <entity2 rdf:resource="http://a.com/ontology#Woods"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Habitats"/> <entity2 rdf:resource="http://a.com/ontology#SnowField"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Habitats"/> <entity2 rdf:resource="http://a.com/ontology#Habitats"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Apoda"/> <entity2 rdf:resource="http://a.com/ontology#Amphibian"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://a.com/ontology#Apoda"/> <entity2 rdf:resource="http://a.com/ontology#Apoda"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure> <relation>=</relation></Cell></map></rdf:RDF>

Tabela A.6 - Alinhamento de referência para o cenário de uso do *WordNet*.

Apêndice B

Neste apêndice, temos as ontologias do domínio de educação usadas no experimento do de conhecimento externo. As ontologias comparadas foram:



Figura B.1 - Ontologia de entrada 1 do primeiro experimento de conhecimento externo usando o domínio de educação

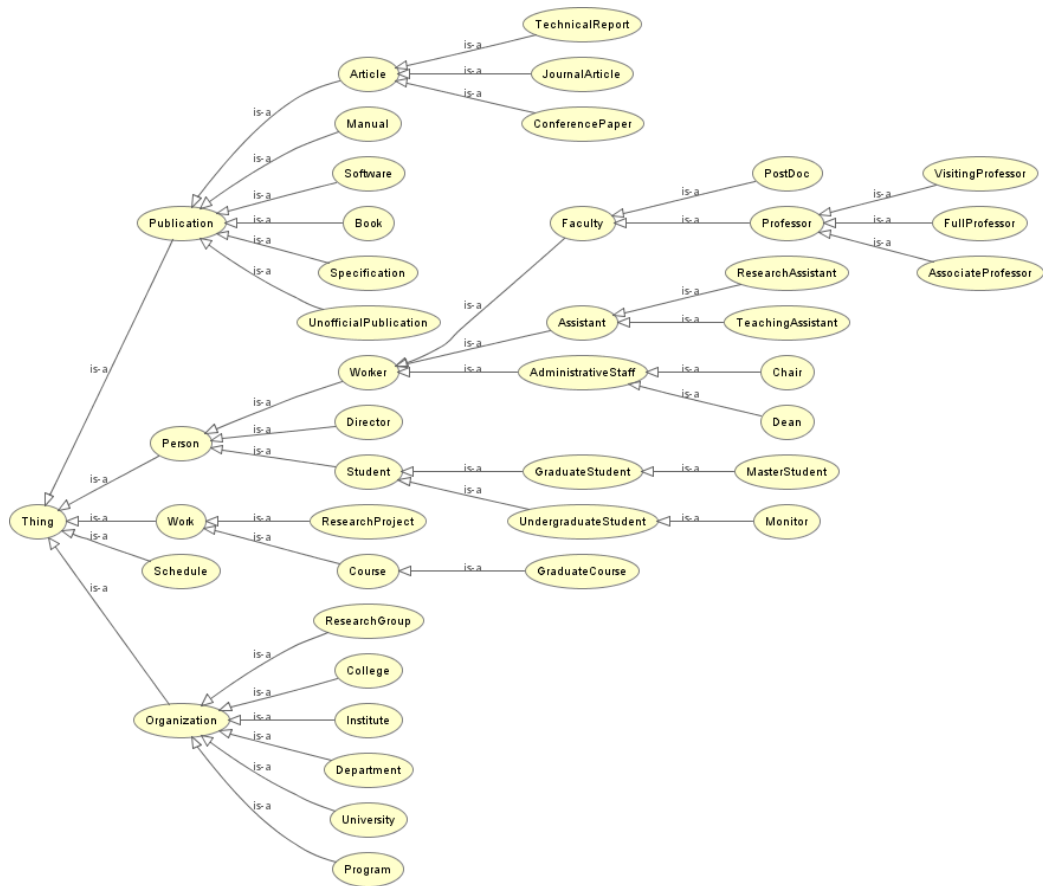


Figura B.2 - Ontologia de entrada 2 do primeiro experimento de conhecimento externo usando o domínio de educação

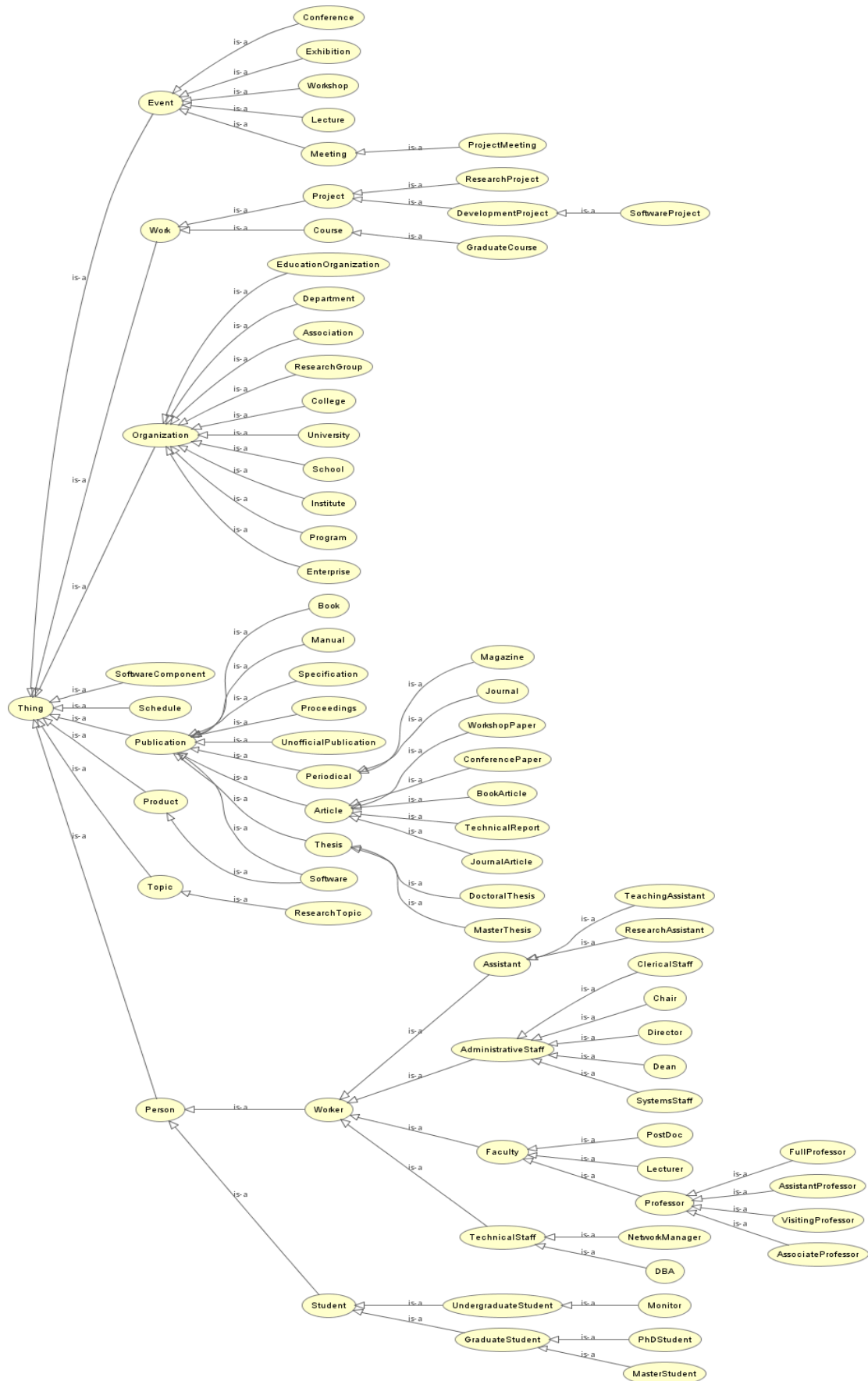


Figura B.3 - Ontologia de domínio do primeiro experimento de conhecimento externo usando o domínio de educação

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns="http://knowledgeweb.semanticweb.org/heterogeneity/alignment"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#">
  <Alignment>
    <xml>yes</xml>
    <level>0</level>
    <type>**</type>
    <onto1>http://swrc.ontoware.org/ontology/portal</onto1>
    <onto2>http://www.lehigh.edu/~zhp2/univ-bench.owl</onto2>
    <uri1>http://swrc.ontoware.org/ontology/portal</uri1>
    <uri2>http://www.lehigh.edu/~zhp2/univ-bench.owl</uri2>
    <map>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#AdministrativeStaff"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Worker"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Director"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Dean"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Chair"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Faculty"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#AdministrativeStaff"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Assistant"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Article"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Publication"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#JournalArticle"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#ConferencePaper"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#TechnicalReport"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Book"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.3</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Manual"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#UnofficialPublication"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Specification"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Book"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
      <Cell>
        <entity1 rdf:resource="http://swrc.ontoware.org/ontology/portal#Article"/>
        <entity2 rdf:resource="http://www.lehigh.edu/~zhp2/univ-bench.owl#Software"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure>
        <relation>=</relation></Cell>
    </map>
  </Alignment>
</rdf:RDF>

```


[illegible]

[illegible]

[illegible]

Apêndice C

Neste apêndice, temos as ontologias do domínio de animais usadas no segundo experimento de conhecimento externo. São ontologias diferentes das usadas nos experimentos com o *WordNet* mas também tratam do domínio de animais. As ontologias comparadas são apresentadas a seguir (a ontologia de domínio foi dividida em partes devido ao seu tamanho):



Figura C.1 - Ontologia de entrada 1 do segundo experimento de conhecimento externo usando o domínio de animais

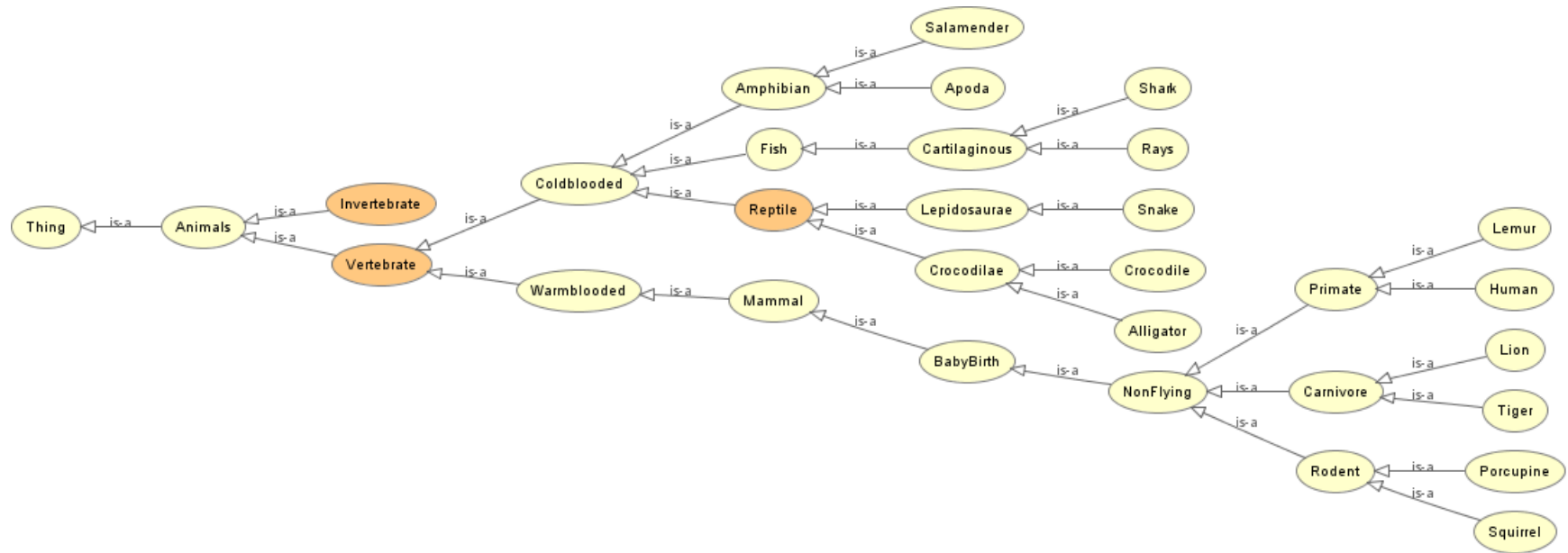


Figura C.2 - Primeira parte da ontologia de entrada 2 do segundo experimento de conhecimento externo usando o domínio de animais



Figura C.3 - Segunda parte da ontologia de entrada 2 do segundo experimento de conhecimento externo usando o domínio de animais

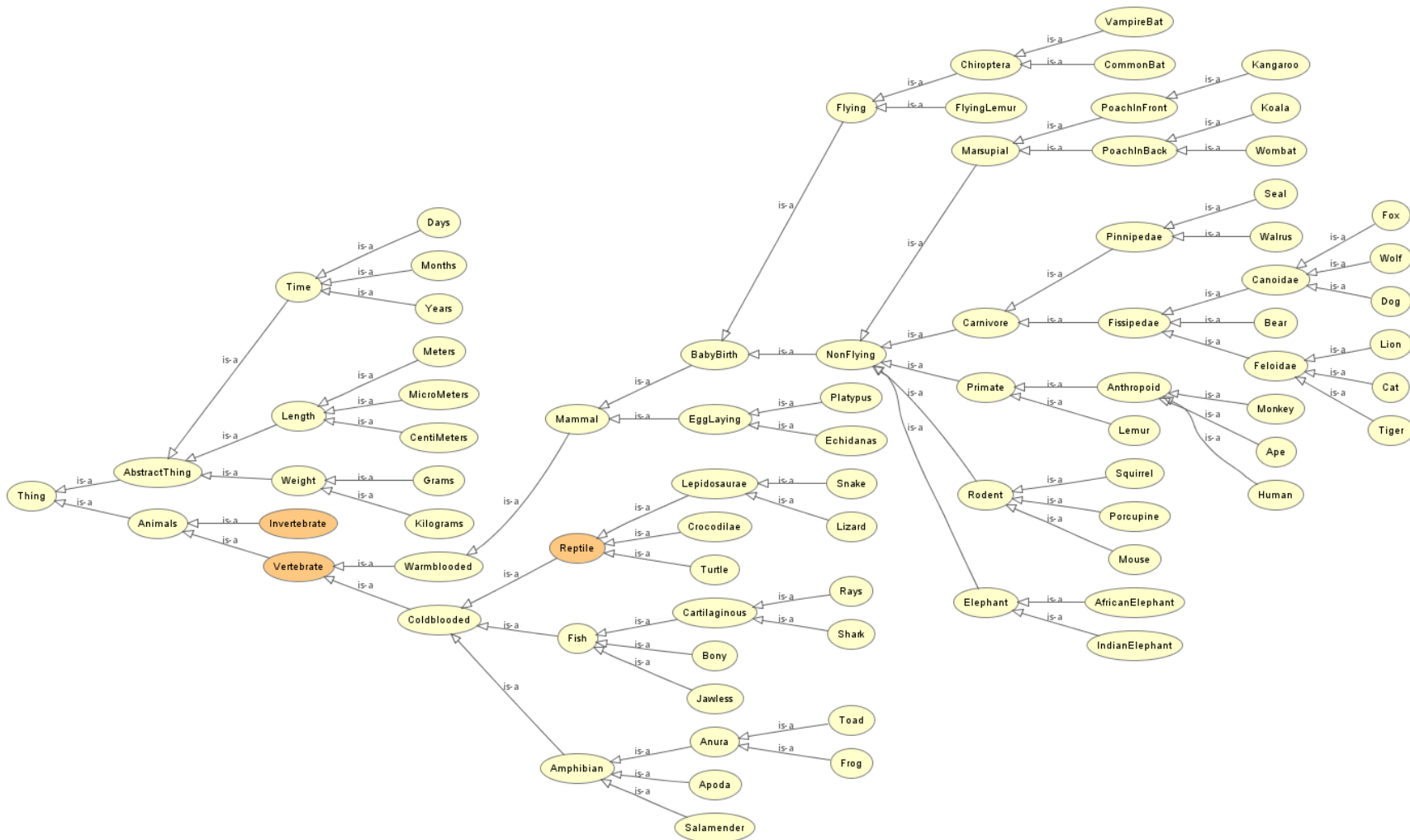


Figura C.4 - Primeira parte da ontologia de domínio usada no segundo experimento de conhecimento externo sobre o domínio de animais

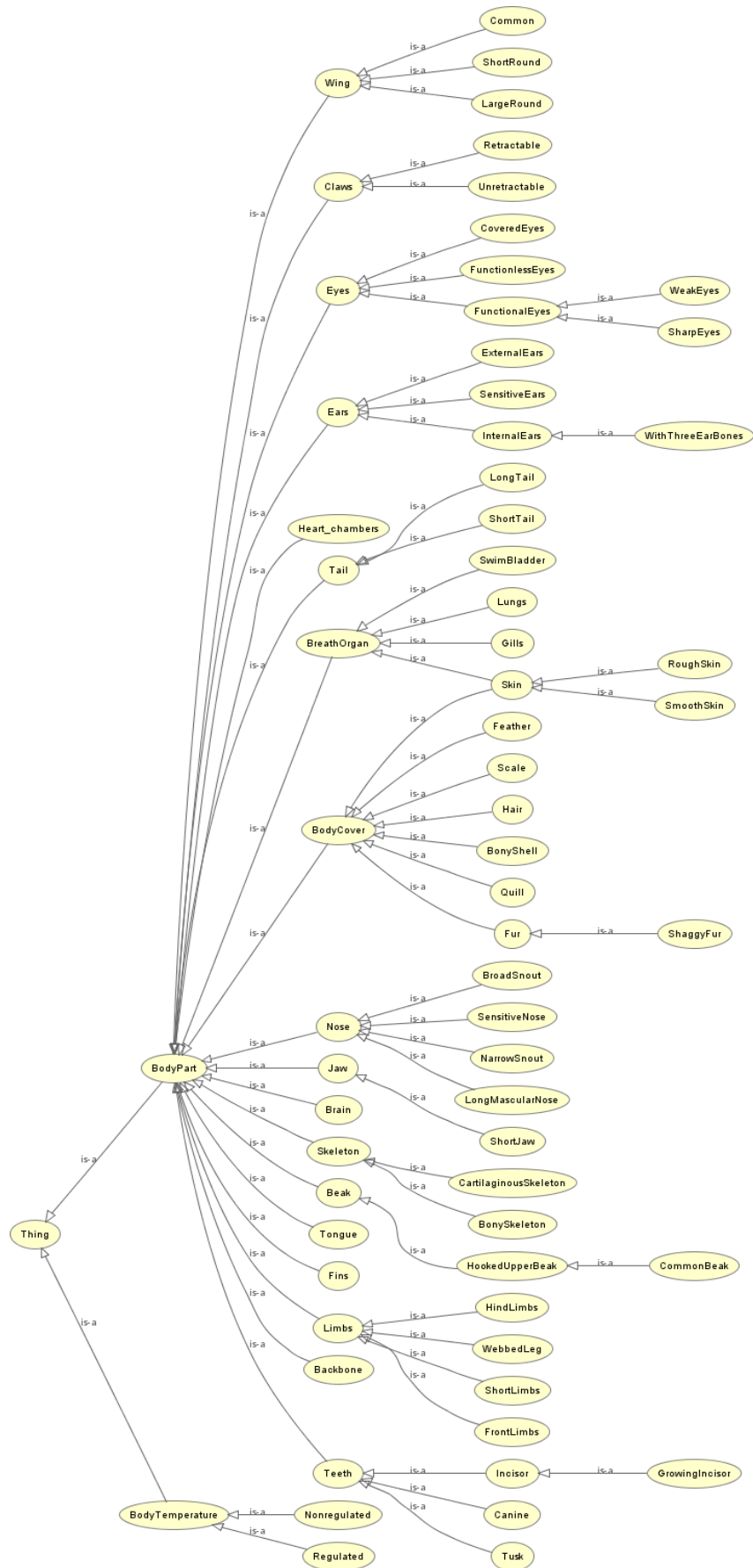


Figura C.5 - Segunda parte da ontologia de domínio usada no segundo experimento de conhecimento externo sobre o domínio de animais



Figura C.6 - Terceira parte da ontologia de domínio usada no segundo experimento de conhecimento externo sobre o domínio de animais

[illegible]

[illegible]

[illegible]

[illegible]

<Cell>	<measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://www.owl-ontologies.com/Ontology1227100040.owl#Wing"/> <entity2 rdf:resource="http://a.com/ontology#Wing"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://www.owl-ontologies.com/Ontology1227100040.owl#Wing"/> <entity2 rdf:resource="http://a.com/ontology#BodyPart"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://www.owl-ontologies.com/Ontology1227100040.owl#Wing"/> <entity2 rdf:resource="http://a.com/ontology#LargeRound"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://www.owl-ontologies.com/Ontology1227100040.owl#Wing"/> <entity2 rdf:resource="http://a.com/ontology#Common"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://www.owl-ontologies.com/Ontology1227100040.owl#Wing"/> <entity2 rdf:resource="http://a.com/ontology#ShortRound"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7999999999999999</measure> <relation>=</relation></Cell>
<Cell>	<entity1 rdf:resource="http://www.owl-ontologies.com/Ontology1227100040.owl#Wing"/> <entity2 rdf:resource="http://a.com/ontology#Skeleton"/> <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.7</measure> <relation>=</relation></Cell></map></Alignment></rdf:RDF>

Tabela C.7 - Alinhamento de referência para o segundo experimento de conhecimento externo sobre o domínio de animais

Apêndice D

Neste apêndice temos as ontologias do domínio de educação usada no experimento de *feedback* do usuário. Todas as ontologias estão listadas a seguir:



Figura D.1 -Ontologia 1 do cenário de Educação para o experimento de histórico

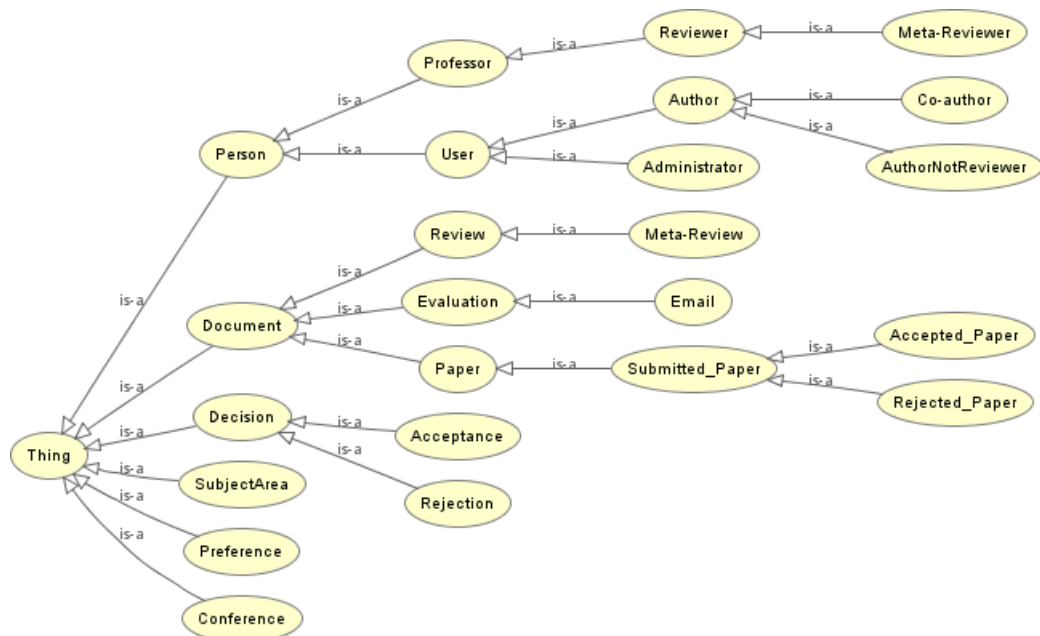


Figura D.2 -Ontologia 2 do cenário de Educação para o experimento de histórico

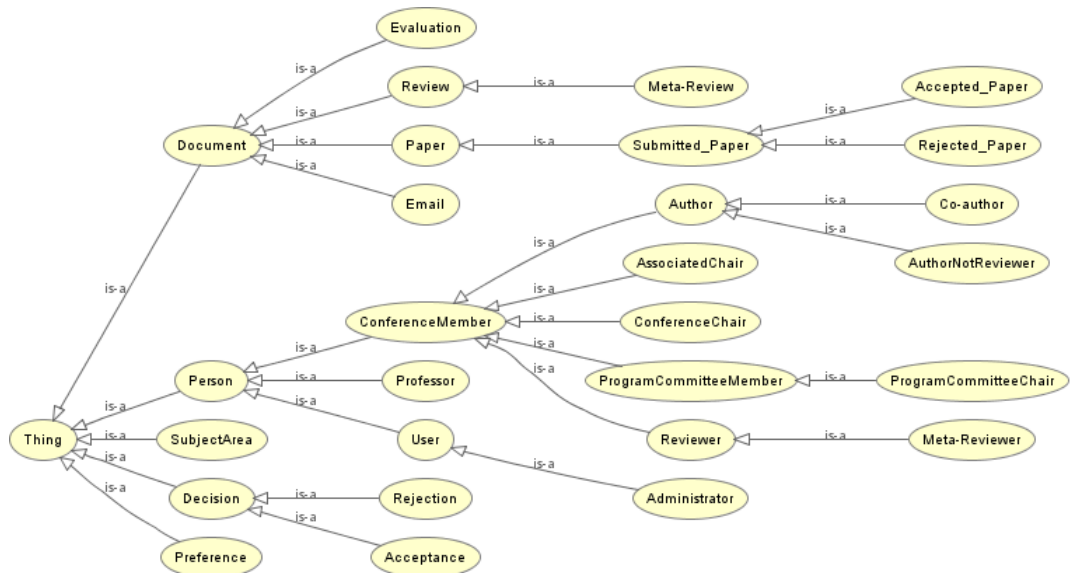


Figura D.3 -Ontologia 3 do cenário de Educação para o experimento de histórico

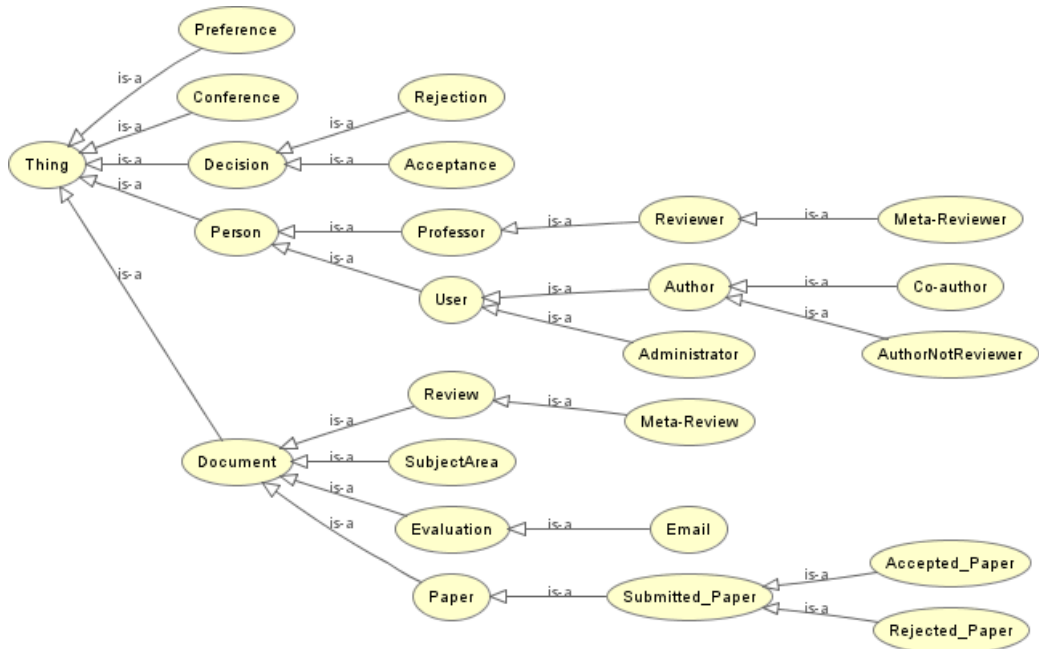


Figura D.4 -Ontologia 4 do cenário de Educação para o experimento de histórico



Figura D.5 -Ontologia 5 do cenário de Educação para o experimento de histórico



Figura D.6 -Ontologia 6 do cenário de Educação para o experimento de histórico

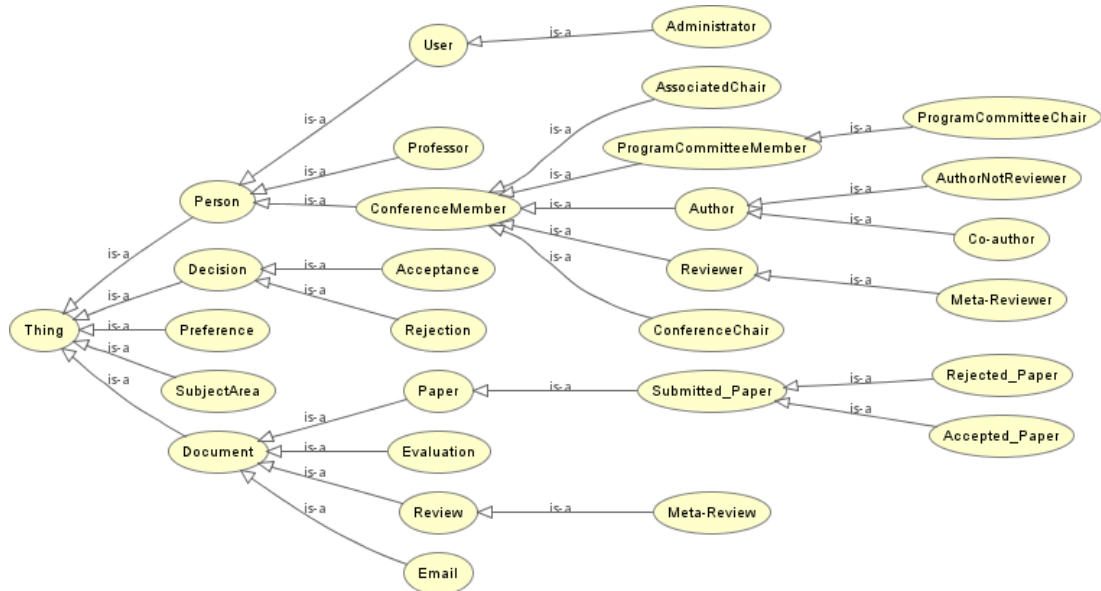


Figura D.7 -Ontologia 7 do cenário de Educação para o experimento de histórico

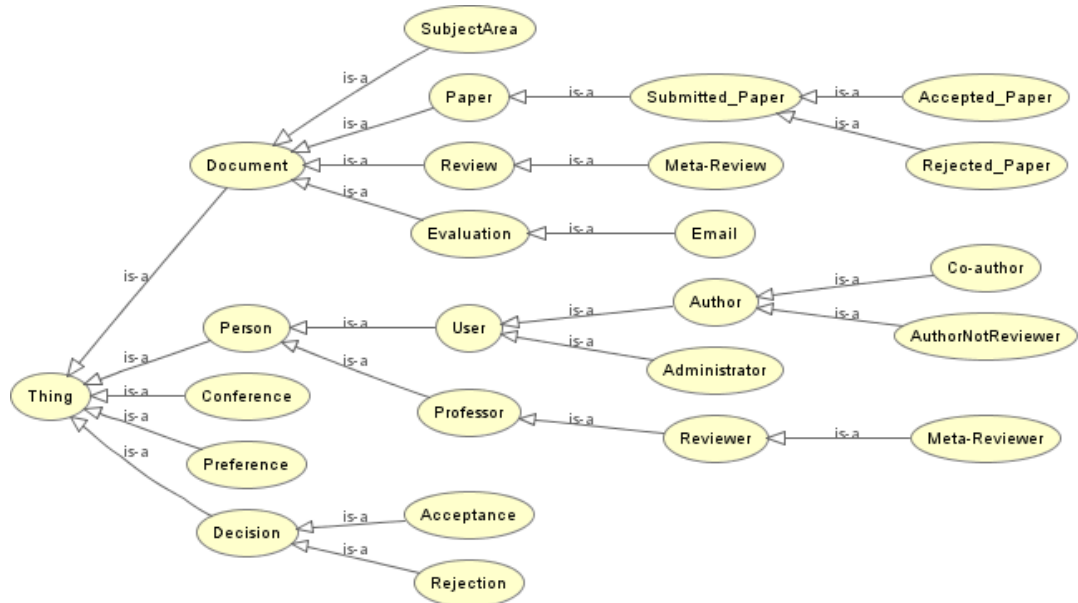


Figura D.8 -Ontologia 8 do cenário de Educação para o experimento de histórico

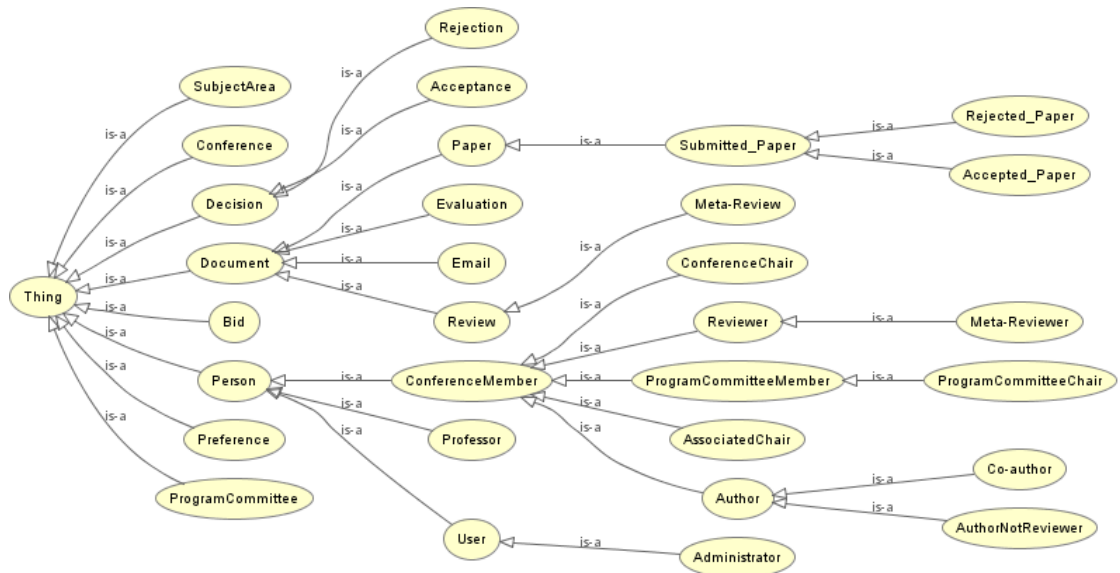


Figura D.9 -Ontologia 9 do cenário de Educação para o experimento de histórico

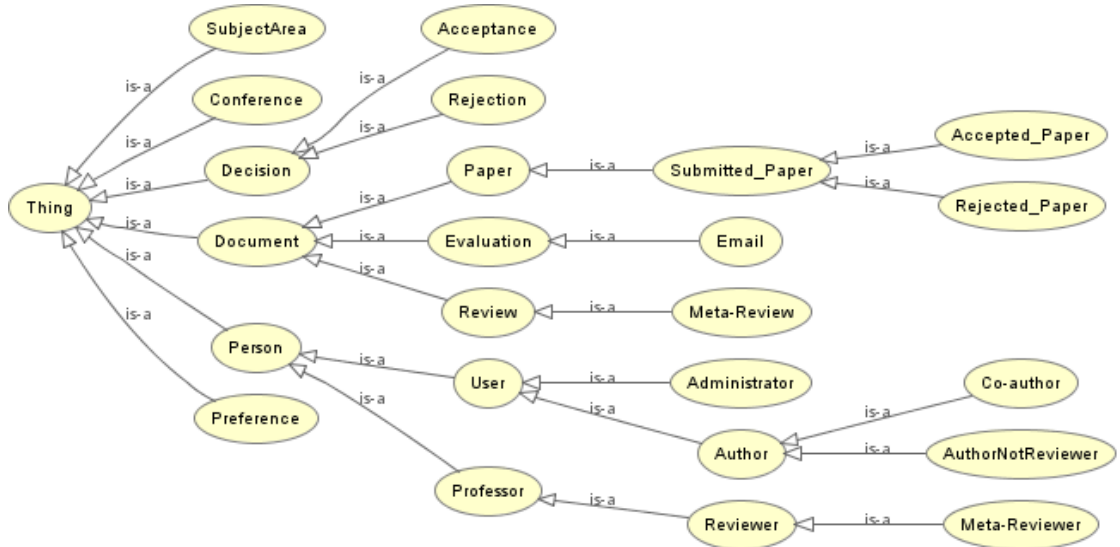


Figura D.10 -Ontologia 10 do cenário de Educação para o experimento de histórico

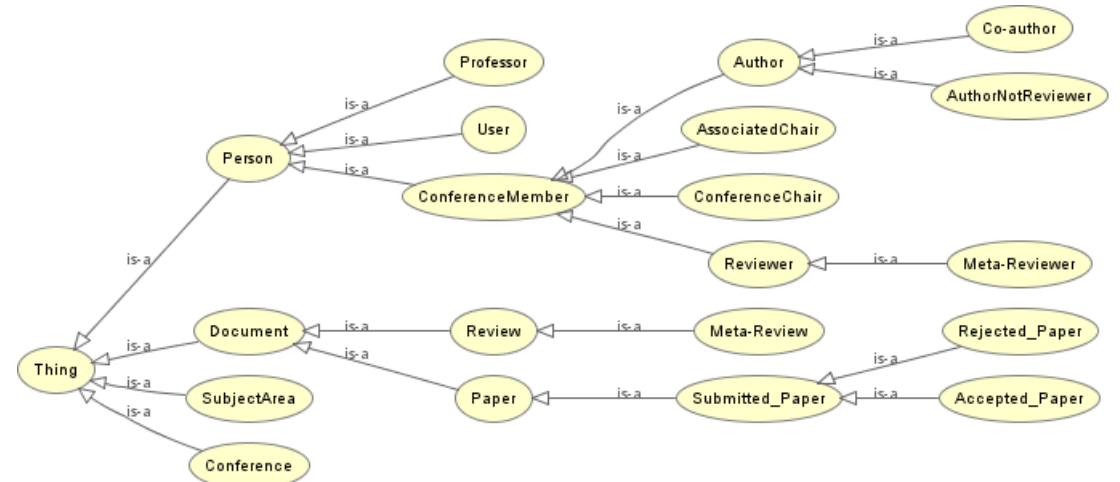


Figura D.11 -Ontologia 11 do cenário de Educação para o experimento de histórico

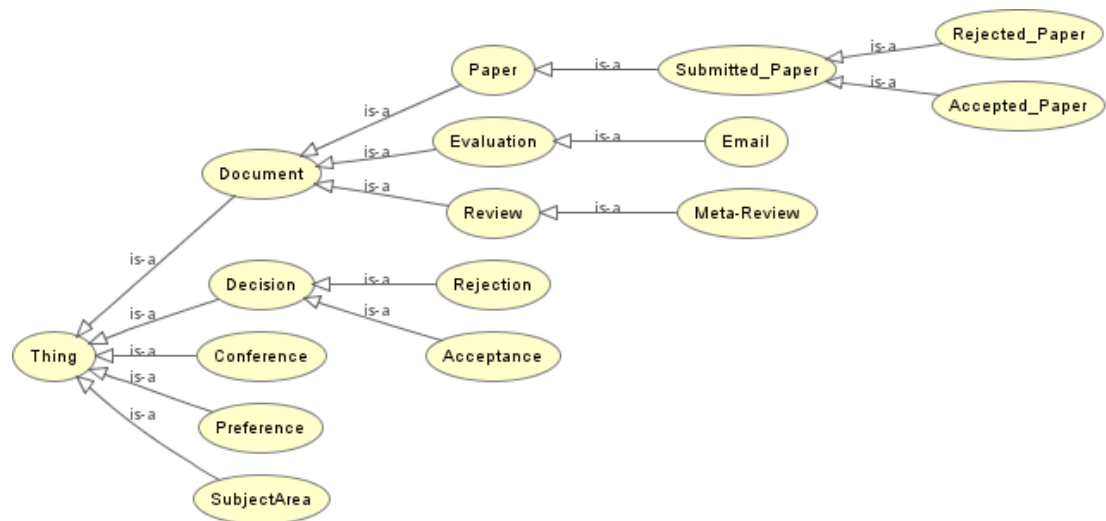


Figura D.12 -Ontologia 12 do cenário de Educação para o experimento de histórico

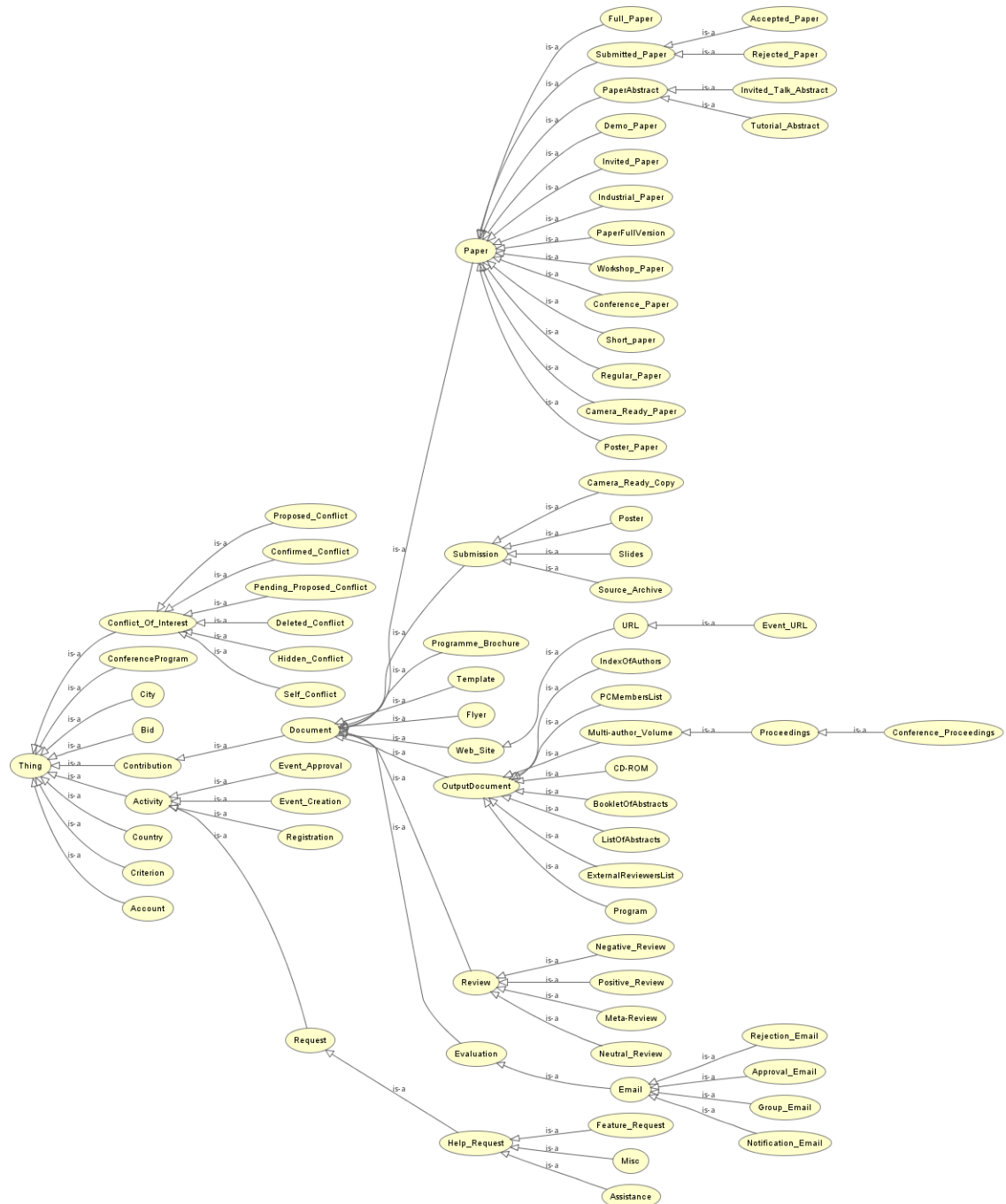


Figura D.13 -Primeira parte da ontologia de domínio do cenário de Educação para o experimento de histórico

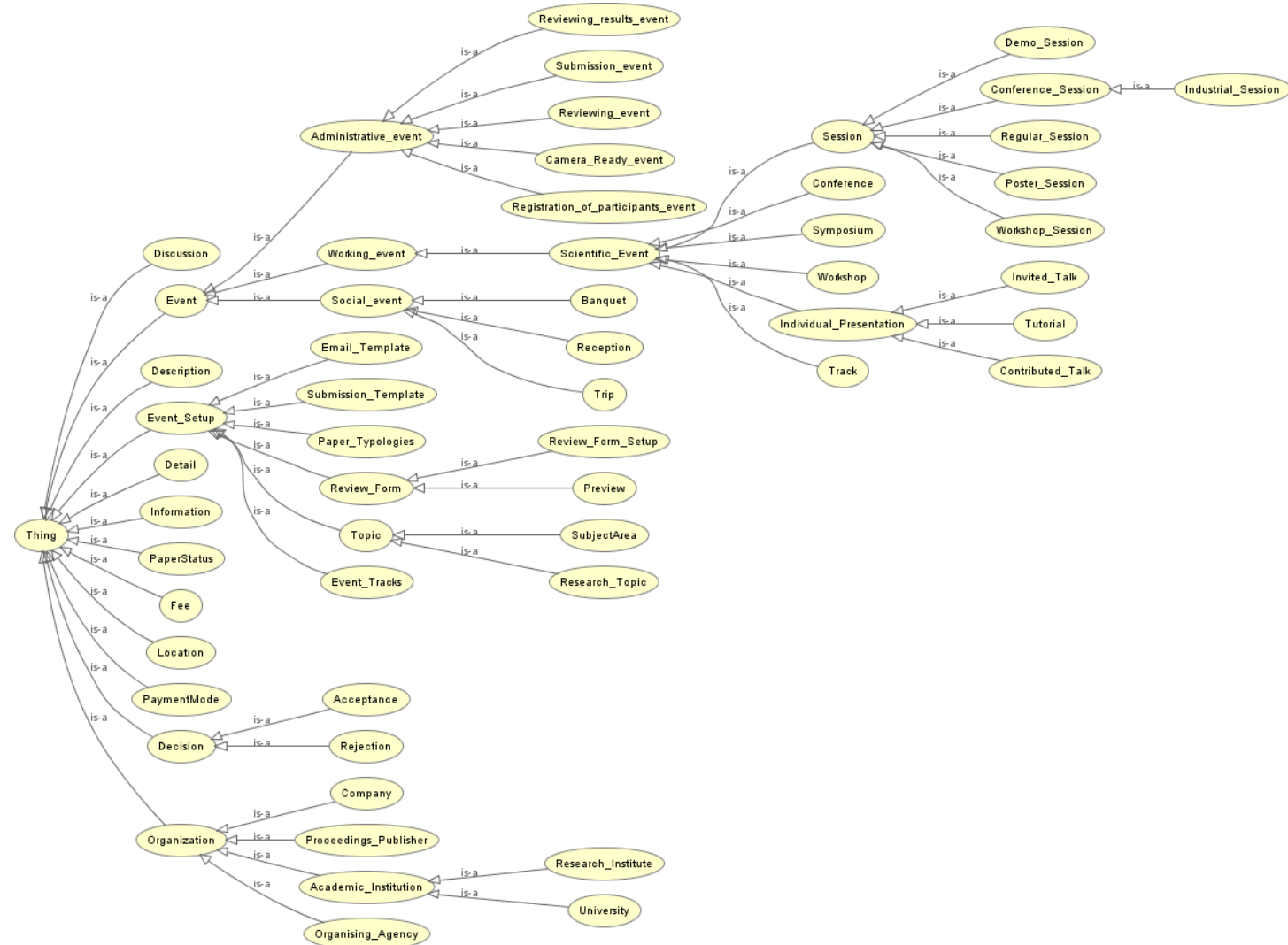


Figura D.14 -Segunda parte da ontologia de domínio do cenário de Educação para o experimento de histórico

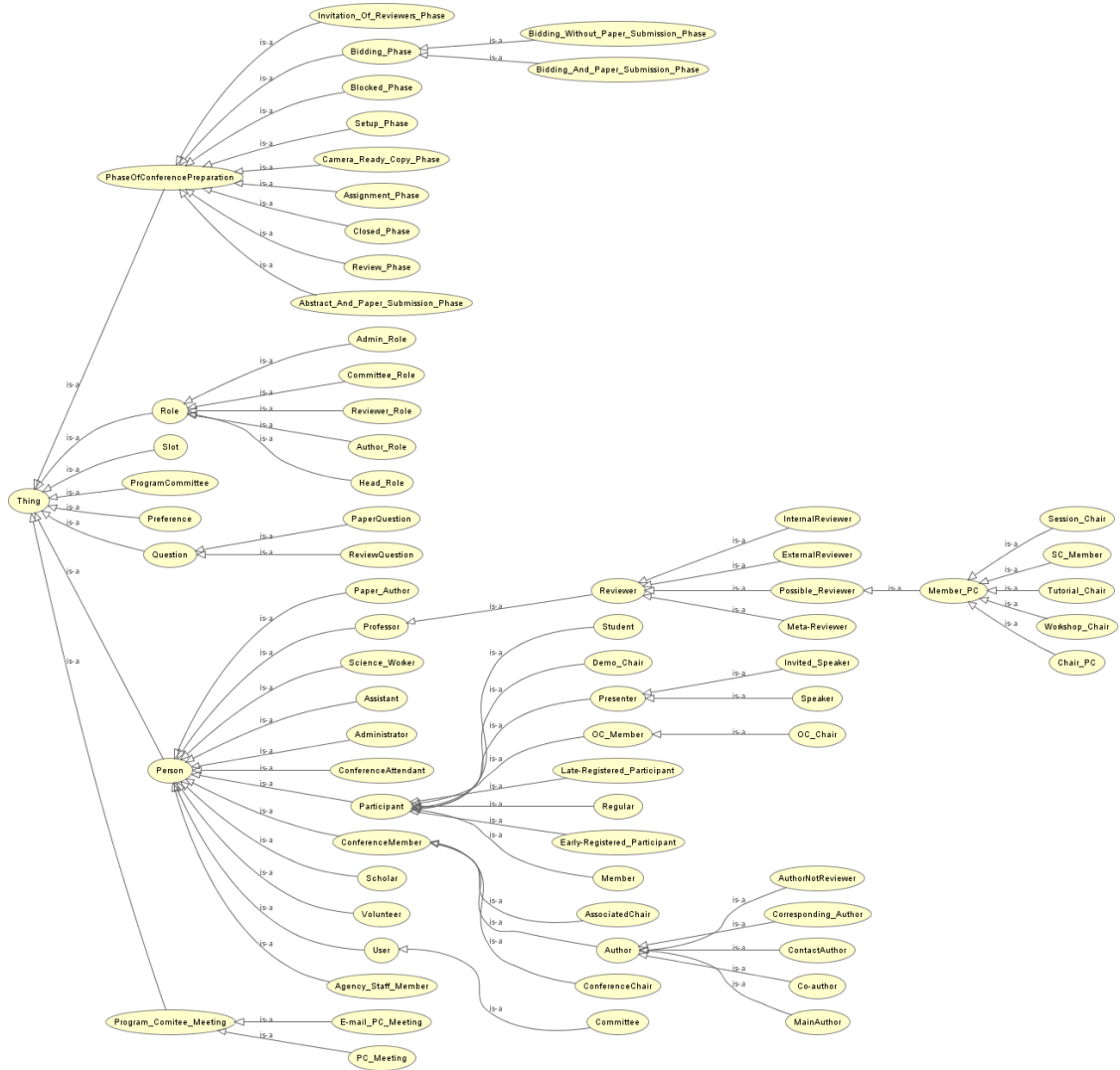


Figura D.15 -Terceira parte da ontologia de domínio do cenário de Educação para o experimento de histórico