Utilização de Ontologias para Busca em um Sistema Colaborativo de Imagens Arquitetônicas

Texto para Exame de Qualificação

Mestrado em Ciência da Computação

Marisol Solis Yucra

Orientadora: Profa. Dra. Renata Wassermann

A autora recebe auxílio financeiro do CNPq

São Paulo, 23 de setembro de 2014

Resumo

SOLIS YUCRA, MARISOL **Utilização de Ontologias para Busca em um Sistema Colaborativo de Imagens Arquitetônicas**. 2014. Dissertação (Mestrado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2014.

"Arquigrafia"é um sistema colaborativo para compartilhamento de imagens arquitetônicas, no qual os usuários fazem o upload de imagens arquitetônicas e registram informação relacionada a cada imagem baseado em etiquetas(tags), uma vez registradas no sistema estas são compartilhadas. O problema atual é a performance do motor de busca, que devolve imagens não relevantes à consulta dada. Isto ocorre porque o sistema permite que os usuários escrevam suas consultas utilizando linguagem livre o que pode ocasionar termos ambíguos e que o atual motor de busca não resolve, prejudicando os resultados . Além disso, o volume de dados do sistema mencionado cresce diariamente, fazendo com que a classificação seja essencial.

Por este motivo, neste trabalho apresentamos uma abordagem de pesquisa que ajudará a melhorar os resultados do motor de busca, fazendo uso de uma ontologia e de modelos da recuperação da informação.

Para a construção da ontologia utiliza-se informação registrada no sistema Arquigrafia, como as etiquetas, os títulos, a localização e os autores de cada imagem, conjuntamente com o histórico de consultas. Assim, por meio da ontologia pode-se obter termos similares aos termos pertencentes à consulta solicitada pelo usuário, que serão adicionados à consulta inicial, transformando-se numa consulta expandida que será utilizada pelo processo da recuperação da informação.

Logo o processo da recuperação da informação devolverá imagens relacionadas à consulta expandida fazendo uso do modelo de espaço vetorial, que calculará a similaridade entre os termos pertencentes aos documentos que contêm informação das imagens, com os termos pertences à consulta; como resultado desse cálculo serão mostradas as imagens mais relevantes à consulta solicitada.

Palavras-chave: vocabulário-controlado, arquivo de índice invertido, etiquetas, ontologia, consultas expandidas, recuperação da informação.

Sumário

Lista de Figuras

Lista de Algoritmos

Capítulo 1

Introdução

Arquigrafia é um sistema ou ambiente colaborativo atual para compartilhamento de imagens de arquitetura[1]. Neste sistema, cada usuário tem a opção de fazer o upload de imagens e guardar informação. Esta informação é compartilhada com outros usuários, que também pertencem ao sistema, podendo fazer o download das mesmas. Além disso, o usuário pode fazer buscas no sistema para recuperar imagens específicas, porém, na maioria das vezes, a busca retorna conteúdos irrelevantes, sendo um problema atual no desempenho do motor de busca.

Este problema acontece por causa do método de consulta utilizado pelo usuário, que é livre para escrever os termos, possibilitando erros e termos ambíguos não entendidos pelo sistema atual. Diariamente, o volume de informação aumenta, incluindo várias etiquetas para cada imagem (tags), complicando assim o desempenho do motor de busca. Para simplificar este problema, é necessário classificar, organizar e tratar esta informação, fazendo com que as consultas sejam não-ambíguas. Além disso, os termos da consulta do usuário podem conter erros ou alguma palavra chave importante, nesse caso utilizar técnicas de recuperação de informação (RI) e a ontologia para consultas expandidas, ajudaria a resolver este problema e a melhorar a qualidade nos resultados da busca.

Assim, este trabalho propõe a construção de uma ontologia de aplicação para o domínio de arquitetura voltado ao sistema Arquigrafia, fazendo uso das etiquetas relacionadas às imagens, do vocabulário de arquitetura da USP, da informação relevante do sistema Arquigrafia e do histórico de consultas, para que junto com a aplicação do processo de recuperação de informação, as consultas devolvam as imagens mais relevantes para o usuário.

1.1 Motivação

O motor de busca atual do Sistema Colaborativo Arquigrafia não permite obter resultados relevantes para consultas em linguagem natural.

Por isto, o desenvolvimento de um sistema de busca baseado em ontologias e RI, possibilitaria a obtenção de resultados relevantes ao usuário, abordando o problema apresentado pelo sistema atual.

1.2 Objetivos

Objetivos gerais:

- Construir uma ontologia para um sistema de compartilhamento de imagens arquitetônicas utilizando o vocabulário controlado de arquitetura da USP e as etiquetas criadas pelo usuário;

[1] Arquigrafia: http://www.arquigrafia.org.br/18/?firstTime=0

- Implementar um método para recuperação de imagens relevantes do sistema de compartilhamento de imagens de Arquigrafia.

Objetivos Específicos:

- Utilizar a ontologia para obter consultas expandidas para serem utilizadas pelo processo de recuperação de imagens.
- Avaliar a ontologia e o método de recuperação, comparando os resultados do método utilizado com a obtenção de resultados de maneira manual.
- Testar o desempenho da busca de imagens, comparando os resultados da busca com e sem a ontologia.

1.3 Organização do Trabalho

No Capítulo 2, apresentamos os conceitos. No Capítulo 3 apresentamos a metodologia que está sendo desenvolvida No Capítulo 4 apresentamos os resultados preliminares. Finalmente, no Capítulo 5 apresentamos o trabalho futuro.

Capítulo 2

Conceitos

Neste capítulo são apresentadas algumas definições e conceitos fundamentais e definições de ontologias, além de ferramentas, linguagens para ontologia, e métodos de recuperação de informação utilizadas ao longo do trabalho.

2.1 Ontologia

Existem várias definições para ontologias. Segundo Gruber (1995) uma ontologia é uma especificação de uma conceitualização, isto é, é uma descrição de conceitos e relacionamentos que existem entre estes conceitos.

Para Borst (1997) uma ontologia é definida como uma especificação formal e explícita de uma conceitualização compartilhada. A especificação formal quer dizer algo que é legível para os computadores; explícita são os conceitos, propriedades, relações, funções, restrições e axiomas explicitamente definidos; a conceitualização representa um modelo abstrato de algum fenômeno do mundo real; compartilhada significa conhecimento consensual.

Os componentes básicos de uma ontologia são conceitos (organizados em uma taxonomia), relações que representam o tipo de interação entre os conceitos do domínio, axiomas usados para modelar sentenças sempre verdadeiras e instâncias que são utilizadas para representar indivíduos descritos pelos conceitos e relacionamentos Almeida e Bax (2003). O conjunto de hipóteses na qual consiste uma ontologia tem a forma da teoria de lógica de primeira ordem, na qual as palavras do vocabulário aparecem como conceitos unários de relações binárias Guarino (1998) Daconta *et al.* (2003)

Assim as ontologias permitem o compartilhamento e reutilização do conhecimento, porém é importante que os conceitos tenham uma especificação formal.

2.2 Tipos de Ontologia

Existem várias formas de classificar as ontologias, para este trabalho, é utilizada a classificação baseada na sua função, definida em cinco categorias Guizzardi (2000):

Ontologias Genéricas: descrevem conceitos amplos, não considerando um problema ou domínio específico.

Ontologias de Domínio: descrevem conceitos e conjuntos de palavras relacionadas a um domínio específico.

Ontologias de Tarefas: não dependem de um domínio particular, descrevem tarefas que podem contribuir na solução de problemas.

Ontologias de Aplicação: descrevem conceitos e vocabulários de um domínio e uma tarefa específica.

Ontologias de Representação: explicam os conceitos que fundamentam o formalismo da representação do conhecimento.

2.3 Linguagens de representação

Existem várias linguagens de representação de ontologias, que estão baseadas na sintaxe do XML(Extensible Markup Language):

2.3.1 RDF (Resource Description Framework)

Para McBride (2004) o RDF é uma recomendação da W3C que define uma linguagem para descrever recursos e foi projetada para descrever recursos web tais como páginas web.
RDF amplia a estrutura de enlaces da web para usar URIs e nomear a relação entre as coisas, assim como as duas extremidades dos enlaces (usualmente referido como uma "tripla").
Cada tripla pode ser vista como uma estrutura definindo a relação entre dois recursos.

Esta estrutura de ligação forma um grafo dirigido e etiquetado, no qual as arestas representam o nome da ligação entre dois recursos, representados pelos nós do grafo. A estrutura de um grafo de RDF[1] é um conjunto de triplas, que consiste de um sujeito, predicado e objeto. E pode ser visualizado como um diagrama de nós e um arco direcionado.

Por exemplo a figura 2.1 mostra uma tripla em RDF.

Figura 2.1: *Exemplo de grafo em RDF*

E o fragmento de código em RDF:

```
<NamedIndividual rdf:about="&ontoArquigrafia;AfaloRoberto">
       <rdf:type rdf:resource="&ontoArquigrafia;AutorDaObra"/>
</NamedIndividual>
```

Usando este modelo simples, RDF permite a intercalação de dados estruturados e semi-estruturados, expostos, e compartilhados entre diferentes aplicações.

2.3.2 RDFS (Resource Description Framework Schema)

RDF Schema (RDFS)[2] é uma extensão semântica de RDF. Ela fornece mecanismos para descrever grupos de recursos relacionados e a relação entre estes recursos. Eles são utilizados para determinar as características de outros recursos, tais como domínio e imagem das propriedades. RDF Schema fornece um vocabulário de modelagem de dados para dados RDF e descreve as propriedades em termos da classe e qual é o tipo para o respectivo valor da propriedade, para a qual se aplicam os mecanismos do domínio e intervalo.

Por exemplo a figura 2.2 mostra um grafo em RDF e em RDFS.

Fragmento de código para o grafo mostrado na figura 2.2 :

[1]Grafo de RDF: http://www.w3.org/TR/rdf11-concepts/
[2]RDFs: http://www.w3.org/TR/rdf-schema/

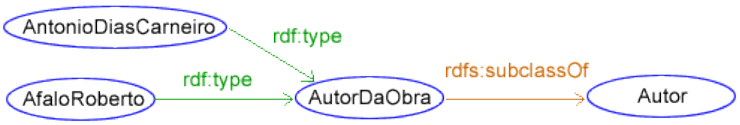


Figura 2.2: *Exemplo de um grafo em RDFs e RDF*

```
<rdf:RDF xmlns="http://www.w3.org/2002/07/owl#"
     xml:base="http://www.w3.org/2002/07/owl"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:ontoArquigrafia="http://www.semanticweb.org/ontologies/2014/
     5/ontoArquigrafia#">
    <Ontology rdf:about="http://www.semanticweb.org/ontologies/2014/
    5/ontoArquigrafia"/>

<Class rdf:about="&ontoArquigrafia;AutorDaObra">
       <rdfs:subClassOf rdf:resource="&ontoArquigrafia;Autor"/>
</Class>
<NamedIndividual rdf:about="&ontoArquigrafia;AfaloRoberto">
       <rdf:type rdf:resource="&ontoArquigrafia;AutorDaObra"/>
</NamedIndividual>
<NamedIndividual rdf:about="&ontoArquigrafia;AntonioDiasCarneiro">
       <rdf:type rdf:resource="&ontoArquigrafia;AutorDaObra"/>
</NamedIndividual>

</rdf:RDF>
```

2.3.3 Web Ontology Language 2 (OWL 2)

OWL 2[3] é uma linguagem de representação de conhecimento desenvolvida no âmbito do W3C - World Wide Web Consortium que descreve um estado de coisas de uma maneira lógica. Esta linguagem é projetada para formular, mudar e raciocinar com conhecimento a respeito de um domínio de interesse. Algumas noções são explicadas para entender como o conhecimento é representado em OWL 2:

- Entidades: São elementos usados para referenciar objetos do mundo real.
- Axiomas: São enunciados básicos que uma ontologia em OWL expressa.
- Expressões: São as combinações de entidades para formar descrições complexas.

OWL 2 fornece três perfis[4], os quais têm fragmentos lógicos com a capacidade expressiva adequada para tarefas distintas e raciocínio eficiente.

- OWL 2 EL: Trabalha com aplicações que utilizam grandes ontologias que têm grande número de classes ou propriedades. Possui checagem de consistência e de instâncias em tempo polinomial.

[3]OWL 2: http://www.w3.org/TR/owl2-primer/#What_is_OWL_2.3F
[4]Perfis OWL 2: http://www.w3.org/TR/owl2-primer/#OWL_2_Profiles

- OWL 2 QL: Suporta as consultas conjuntivas e utiliza a tecnologia padrão de bancos de dados relacionais. Também captura muitas características utilizadas em RDFS das propriedades inversas e das sub-propriedades hierárquicas.
- OWL 2 RL: É dirigida para aplicações que exigem raciocínio escalável sem sacrificar muito o poder expressivo. Esta linguagem é ideal para enriquecer dados RDF, especialmente quando os dados deveriam ser manipulados por regras adicionais.

2.3.4 Componentes e elementos da Ontologia em OWL 2

Todos os componentes da ontologia podem ser representados por diferentes sintaxes, mas para entender estes componentes, serão apresentados exemplos na sintaxe RDF/XML.
Uma ontologia OWL 2 é composta pelos seguintes elementos:

1. **Classes**: Elas tem uma representação concreta de um conceito ou entidade, e elas podem conter um conjunto de indivíduos, os quais compartilham algumas propriedades. Também existem hierarquias de classes que compartilham a relação *é um*, quer dizer que existe uma superclasse e uma subclasse na hierarquia. No seguinte exemplo definimos a classe *"Artigo"* como subclasse da classe *"Publicacao"*.

```
<owl:Class rdf:about="&OntoEducacional;Artigo">
    <rdfs:subClassOf rdf:resource="&OntoEducacional;Publicacao"/>
</owl:Class>
```

   Outro exemplo é a classe *"ArtigoRevista"* como subclasse de *"Artigo"*

```
<owl:Class rdf:about="&OntoEducacional;ArtigoRevista">
        <rdfs:subClassOf rdf:resource="&OntoEducacional;Artigo"/>
    </owl:Class>
```

2. **Propriedades**: É a relação que existe entre objetos do OWL 2. Existem dois tipos de propriedades e o comportamento deles é restringido pela especificação do domínio (origem) e imagem (escopo).

   2.1 Propriedade de tipo de dados: São relações entre a instância de uma classe e um literal ou tipo predefinido. Por exemplo temos a propriedade *"TemDataDePublicacao"* que relacionará o domínio que uma instância da alguma classe com o literal *"String"* que é a imagem.

```
<owl:DatatypeProperty rdf:about="&OntoEducacional;TemDataDePublicacao">
        <rdfs:range rdf:resource="&xsd;string"/>
    </owl:DatatypeProperty>
```

   2.2 Propriedade de objeto: É uma relação binária entre instâncias de duas classes. Por exemplo temos a propriedade *"publicadoEmRevista"* que relaciona a instância da classe *"ArtigoRevista"* que é o domínio com a instância da classe *"Revista"* que é a imagem. Além de existir uma hierarquia nas propriedade de objeto, neste exemplo podemos ver que a propriedade *"publicadoEmRevista"* é sub-propriedade da propriedade *"publicadoEm"*.

```
<owl:ObjectProperty rdf:about="&OntoEducacional;publicadoEmRevista">
   <rdf:type rdf:resource="&owl;AsymmetricProperty"/>
   <rdf:type rdf:resource="&owl;FunctionalProperty"/>
   <rdf:type rdf:resource="&owl;IrreflexiveProperty"/>
   <rdfs:domain rdf:resource="&OntoEducacional;ArtigoRevista"/>
```

```
        <rdfs:range rdf:resource="&OntoEducacional;Revista"/>
        <rdfs:subPropertyOf rdf:resource="&OntoEducacional;publicadoEm"/>
    </owl:ObjectProperty>
```

Além disso todas as propriedades podem ter as seguintes características:

- Propriedade funcional: Para esta propriedade cada indivíduo pode-se relacionar apenas a um outro indivíduo a partir dela.
  Um exemplo pode ser a propriedade *temMãe* onde um indivíduo A pode se relacionar com a propriedade mencionada uma vez só a um indivíduo B.
- Propriedade funcional inversa: Se uma propriedade é funcional inversa quer dizer que a relação inversa é funcional. Quer dizer que para um indivíduo A só pode haver um outro indivíduo B relacionado a ele através da propriedade.
  Por exemplo se o indivíduo B é *marido* do indivíduo A e o indivíduo C é *marido* do indivíduo A, então podermos inferir que o indivíduo B e o indivíduo C são o mesmo indivíduo.
- Propriedade transitiva: Para esta propriedade, se temos ao indivíduo A relacionado ao indivíduo B, e B relacionado ao indivíduo C, então podemos inferir que o indivíduo A também terá a mesma relação com o indivíduo C.
- Propriedade simétrica: Para esta propriedade, se o indivíduo A se relaciona com o indivíduo B implica que o indivíduo B também se relaciona com o indivíduo A.
- Propriedade assimétrica: Para esta propriedade, temos ao indivíduo A relacionado com o indivíduo B mas B não se relaciona com A.
- Propriedade reflexiva: Nesta propriedade temos a um indivíduo A que se relaciona com ele mesmo.
- Propriedade irreflexiva: Relaciona um indivíduo A com um indivíduo B sempre que B seja diferente de A.

3. **Indivíduos**: São as instâncias de uma classe particular e objetos de um domínio que podem pertencer a mais de uma classe. No seguinte exemplo, temos a instância da classe *"ArtigoRevista"*, identificada por *"InteractionOfProtocols"*.

```
<owl:NamedIndividual rdf:about="&OntoEducacional;InteractionOfProtocols\">
        <rdf:type rdf:resource="&OntoEducacional;ArtigoRevista"/>
        <OntoEducacional:publicadoEmRevista
        rdf:resource="&OntoEducacional;Elsevier"/>
    </owl:NamedIndividual>
```

Em outro exemplo temos a instância da classe *"Revista"*, identificada por *"Elsevier"*.

```
<owl:NamedIndividual rdf:about="&OntoEducacional;Elsevier">
        <rdf:type rdf:resource="&OntoEducacional;Revista"/>
    </owl:NamedIndividual>
```

2.4 Ferramentas para ontologias

Existem várias ferramentas disponíveis para a construção de ontologias, que são úteis para diminuir a complexidade no desenvolvimento das mesmas. Também existem ferramentas e APIs para manipular ou consultar ontologias, com o objetivo de simplificar a integração com outras aplicações.

2.4.1 Ferramenta para Construção

- Protégé: É uma ferramenta que ajuda a elaborar as ontologias, através de uma interface gráfica de edição.

  O Protégé ainda possui características como escalabilidade e extensibilidade por sua arquitetura de plugins, a integração dos serviços que oferece e a interface que mostra ao usuário é de fácil utilização, e permite importar e exportar ontologias em OWL 2.

2.4.2 Ferramenta para Manipulação

Apache Jena[5] é um framework Java de código aberto para a construção de web semântica e aplicações de dados vinculados. O framework é composto de diferentes API's (application programming interface) que interagem em conjunto para processar dados RDF.

Entre as APIs disponíveis temos:

1. RDF API, dispõe de funcionalidades para ler e escrever RDF como XML.
2. Ontology API, possui características de web semântica tais como o raciocínio sobre os dados usando OWL.
3. SPARQL API, dispõe de funcionalidades para formular consultas expressivas sobre dados RDF.
   SPARQL[6] é uma linguagem recomendada pelo W3C para consultas a ontologias.

2.5 Metodologia para o desenvolvimento de ontologias

Atualmente existem muitas metodologias para a construção de ontologias, porém não há um padrão para sua construção. Guizzardi (2000), sugere uma abordagem sistemática para o seu desenvolvimento, a qual aborda as principais características das metodologias mais usadas. Esta abordagem tem um ciclo de vida iterativo que pode ser observado na figura 2.3
A metodologia dada por Guizzardi (2000) tem as seguintes atividades:

1. Identificação de propósitos e especificação de requisitos: Tem como objetivo identificar a competência da ontologia e dizer qual será seu uso e quais seus propósitos, por meio da delimitação do que é relevante para a ontologia. Também identifica-se potenciais usuários da ontologia. Para descrever o uso da ontologia, ela deve de ser capaz de responder a algumas questões de competência.
   Estas questões de competência segundo Grüninger e Fox (1995) devem ser relevantes e formuladas para determinar o escopo de uma ontologia, baseando-se no conhecimento básico do domínio. Além disso, estas perguntas servirão para testar a ontologia posteriormente, pois ela deve ser capaz de responder estas questões de competência.
2. Captura da ontologia: É importante capturar o conjunto de elementos de um domínio que vai ser representado por uma ontologia. Esta captura envolve a identificação e especificação de conceitos (classes), seus relacionamentos, suas propriedades, axiomas e instâncias. A utilização de taxonomias deve ser feita, já que elas também ajudam a entender melhor os conceitos.
3. Formalização da ontologia: Corresponde à especificação da ontologia em uma linguagem.

[5]Jena: https://jena.apache.org/
[6]SPARQL: http://sparql.org/

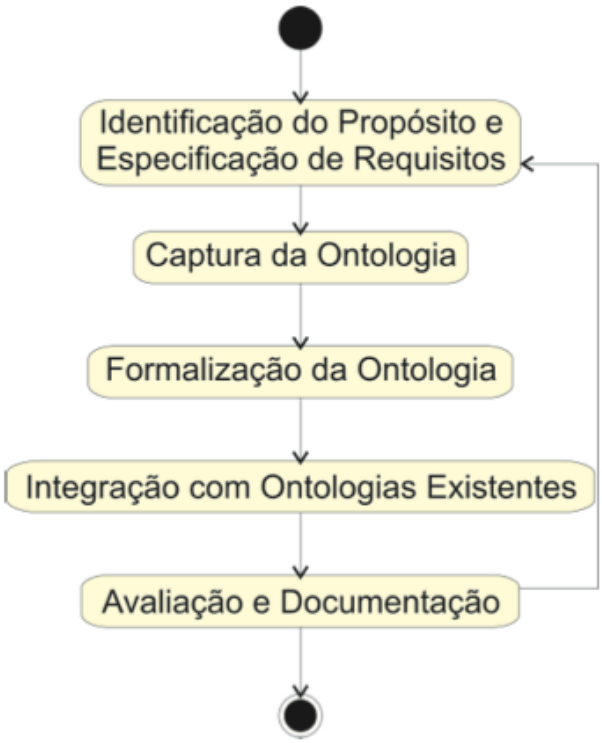


Figura 2.3: *Metodologia proposta por Guizzardi. Fonte da imagem: Morais e Ambrósio (2007)*

4. Integração com ontologias existentes: Integrar a ontologia em questão com outras já existentes, com a intenção de aproveitar os conceitos pertencentes a essa ontologia que já foram estabelecidos. Pode-se desenvolver ontologias com funções modulares visando facilitar sua reutilização e integração.

5. Avaliação: Avaliar uma ontologia significa verificar se ela satisfaz os requisitos definidos em sua construção. Pode-se usar as questões de competência realizadas na fase de Identificação de propósitos e especificação de requisitos.

6. Documentação: Todo o desenvolvimento da ontologia deve ser documentado. Deve-se incluir os propósitos, requisitos e cenários de motivação, as descrições textuais da conceituação, a ontologia formal e os critérios de projeto adotados por Morais e Ambrósio (2007).

2.6 Vocabulário controlado

Um vocabulário controlado pode ser definido como uma lista de termos explicitamente enumerados com o propósito de organizar e representar informação para facilitar a recuperação de informação Mai (2008).

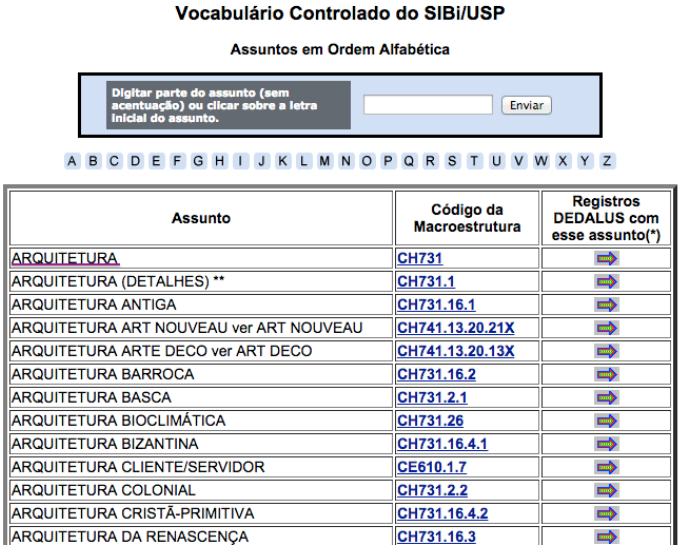
O vocabulário controlado permite uma busca mais fácil em um banco de dados. Dado que se tem diferentes formas de descrever conceitos, o uso de todos estes termos juntos sob uma única palavra ou expressão em um banco de dados torna a busca mais eficiente, pois elimina o trabalho de redução de incerteza. No entanto, chegar a esta eficiência precisa de consistência sob parte do indexamento individual do banco de dados e uso de termos pré-determinados.

O Vocabulário Controlado pode estar na forma de uma simples lista de termos, uma taxonomia ou um extenso tesauro com complexa estrutura hierárquica e diversos tipos de relacionamentos entre os termos N.I.S. e Organization (2005).

Os passos que um desenvolvedor de vocabulário controlado pode seguir estão descritos na literatura e são frequentemente representados em alguma versão da seguinte lista Mai (2008):

1. Analisar a literatura, necessidades, atores, tarefas, domínios, atividades, etc;
2. Coletar, ordenar, e combinar termos;
3. Selecionar descritores e estabelecer relacionamentos;
4. Construir agendas classificadas; e,
5. Preparar o produto final.

O vocabulário controlado da USP[7] é uma lista de termos que serve como um catálogo de terminologia padronizada da linguagem documentária da Universidade de São Paulo. Serve para controlar sinônimos e facilitar a indexação e recuperação da informação. Este vocabulário mostra uma lista de vários assuntos das diferentes áreas de conhecimento ver (figura 2.4). Ao escolher a área de Arquitetura a qual nos interessa, obteremos uma lista hierárquica como vemos na figura 2.5.



Vocabulário Controlado do SIBi/USP

Assuntos em Ordem Alfabética

Digitar parte do assunto (sem acentuação) ou clicar sobre a letra inicial do assunto. Enviar

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Assunto	Código da Macroestrutura	Registros DEDALUS com esse assunto(*)
ARQUITETURA	CH731	
ARQUITETURA (DETALHES) **	CH731.1	
ARQUITETURA ANTIGA	CH731.16.1	
ARQUITETURA ART NOUVEAU ver ART NOUVEAU	CH741.13.20.21X	
ARQUITETURA ARTE DECO ver ART DECO	CH741.13.20.13X	
ARQUITETURA BARROCA	CH731.16.2	
ARQUITETURA BASCA	CH731.2.1	
ARQUITETURA BIOCLIMÁTICA	CH731.26	
ARQUITETURA BIZANTINA	CH731.16.4.1	
ARQUITETURA CLIENTE/SERVIDOR	CE610.1.7	
ARQUITETURA COLONIAL	CH731.2.2	
ARQUITETURA CRISTÃ-PRIMITIVA	CH731.16.4.2	
ARQUITETURA DA RENASCENÇA	CH731.16.3	

Figura 2.4: *Lista de assuntos de vocabulário controlado da USP.*

2.7 Etiquetas (Tags)

Tags são definidas como etiquetas descritivas ou palavras chave que são atribuídas a um objeto. Para nosso trabalho, as etiquetas são definidas como metadados textuais que são atribuídos às imagens.

Tagging (Etiquetagem) é o processo de descrever o que o objeto é, usando etiquetas. Na tendência atual da Web 2.0, tagging é realizada pelos observadores do conteúdo. Estas tags podem ser compartilhadas e usadas por todos os observadores do conteúdo Christiaens (2006).

2.8 Medida de similaridade na recuperação da informação

A ontologia junto à recuperação da informação permite a recuperação de documentos (imagens, vídeos, etc) que contenham no seu conteúdo um termo similar a um dos termos pertencentes à consulta feita pelo usuário. Esta similaridade entre termos é importante para a recuperação de tais documentos, que podem não ter a mesma palavra utilizada na consulta do usuário mas sim alguma palavra similar. Desta maneira pode-se recuperar mais informação relevante para o usuário.

[7]Vocabulário: http://143.107.154.62/Vocab/

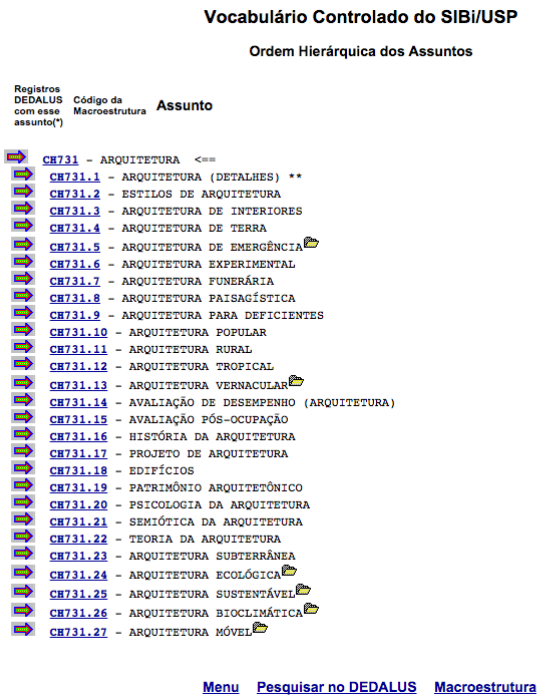


Figura 2.5: *Lista do vocabulário controlado da USP na área da arquitetura.*

Existem diversos trabalhos que usam e definem a medida de similaridade, utilizando os termos de uma hierarquia de conceitos, que para este trabalho serão os termos pertencentes à ontologia.

1. Os autores Schickel-Zuber e Faltings (2007) definem a medida de similaridade para ontologia hierárquicas e a chamam de similaridade baseada na estrutura da ontologia. Eles apontaram que cada termo deveria ser definido como uma função de valor real normalizado para o intervalo de [0,1] e cumprir 3 premissas. A primeira, a pontuação de similaridade depende das características dos termos. Segundo, cada característica contribui independentemente à pontuação. Terceiro, características desconhecidas e improváveis não contribuem à pontuação. Eles utilizam o cálculo de pontuação a priori (a-priori score) de um termo c com n descendentes, calculados como: $APS(c) = 1/n + 2$, que é igual à média de uma distribuição uniforme entre 0 e 1. Reciprocamente o valor mais baixo será encontrado na raiz. Isto significa que quando percorremos a ontologia em direção à raiz, encontramos conceitos mais gerais, e por tanto o APS diminui. Logo, quando comparamos conceitos mais altos na hierarquia, a diferença na pontuação diminui devido ao aumento do número de descendentes.

2. O teorema de similaridade dado por Lin (1998) propõe um método que não leva em consideração apenas os aspectos comuns dos pais dos termos da consulta, mas também o conteúdo de informação associada com os termos da consulta. O método tem três presunções para calcular a similaridade entre dois termos:

   (a) Termos são associados com suas propriedades comuns. Quanto mais propriedades comuns, maior sua similaridade.

   (b) Associados com sua diferença. Quanto mais diferentes são, menor sua similaridade.

   (c) A similaridade alcança o valor máximo quando os termos são exatamente os mesmos.

   Assim baseados nas presunções acima, para os termos, t_i e t_j, sua similaridade é definida como

$$Sim(t_i, t_j) = \frac{2 \log P(t_0)}{\log P(t_i) + \log P(t_j)} \tag{2.1}$$

Na qual t_0 é o ancestral comum mais perto de t_i e t_j, e as probabilidades de ocorrências $P(t_i)$, $P(t_j)$ e $P(t_0)$, que para nosso caso, seriam as frequências dos documentos para t_i, t_j e t_0.

2.9 Recuperação da informação

Permite ao usuário recuperar informação de uma coleção de documentos através de consultas usualmente formatadas, como um conjunto de palavras chaves Baeza-Yates e Ribeiro-Neto (1999).

2.9.1 Processo de Indexação

Usa três componentes principais de entrada Manning *et al.* (2008):

- A coleção de documentos que contêm a informação que vai ser recuperada pelo sistema.
- O dicionário que é uma lista em ordem alfabética das palavras que aparecem no banco de dados.
- O arquivo de índice invertido, que armazena as ocorrências de cada palavra que aparecem no banco de dados. Sua estrutura contém:
  - Token: Que representa a palavra, ou raiz da palavra a indexar.
  - Contagem de documentos: É o número de documentos na qual o token aparece.
  - Contagem de frequência total: O número de vezes que o token aparece no total de documentos.
  - Informação por documento: Uma lista de informações associadas às ocorrências do token em cada documento.
  - Pode-se armazenar além da coleção, informação sobre as palavras utilizadas nos documentos, que pode ser utilizada para enriquecer o processo de recuperação, permitindo inferir relações entre as palavras (por exemplo sinônimos) utilizadas na consulta e as presentes na coleção.

O processo de indexação segue as seguintes etapas:

a) Remoção de afixos: consiste na extração dos radicais das palavras. Após este processo, as variantes das palavras são associadas ao mesmo radical, e só uma entrada é criada no dicionário para estas palavras. Com essa associação as palavras com o mesmo radical recuperam os mesmos resultados.

b) Expansão de termos do índice: no momento da criação do arquivo invertido, os termos não presentes no texto mas que estejam relacionados aos termos que estão no arquivo invertido, podem ser indexados. A relação ou similaridade entre os termos está definida em uma estrutura associada à coleção, que pode ser uma ontologia.

c) Cálculo de pesos: Para o cálculo da relevância de um documento são usados valores que associam cada termo do arquivo invertido aos documentos nos quais aparece. Esses valores podem ser calculados na hora da consulta ou podem ser armazenados no arquivo invertido para evitar sobrecarregar o processo de consulta Jackson e Moulinier (2007).

2.9.2 Processo de Recuperação

Este processo recupera a consulta expandida para calcular a relevância das imagens e a ordenação dos resultados a mostrar de acordo com a relevância para serem apresentados ao usuário Jackson e Moulinier (2007).

O autor Manning *et al.* (2008) apresenta três abordagens para a recuperação da informação: a busca booleana, a recuperação ordenada, e a recuperação probabilística.

- Busca Booleana: O usuário faz a consulta na coleção conectando palavras mediante operadores lógicos (AND, OR, e NOT). É o tipo de busca que os motores de busca frequentemente usam.

- Recuperação ordenada: A ordem é baseada na distribuição de frequência dos termos da consulta na coleção de documentos. Assim as palavras comuns são consideradas com menor importância na ordenação dos resultados. Os documentos e as consultas são tratados como vetores no espaço multi-dimensional. Além disso, se faz o cálculo da similaridade que existe entre a consulta e cada documento, dando um valor de relevância de cada documento com a consulta. Assim os documentos são ordenados em relação a relevância obtida.

- Modelo de espaço Vetorial: Segundo Salton *et al.* (1975) este modelo trabalha com vetores de termos, os quais são palavras ou frases. Cada palavra no vocabulário tem uma dimensão independente em cada vetor de espaço dimensional, assim qualquer texto pode ser representado por um vetor no espaço de alta dimensionalidade. Por exemplo se um termo pertence a um texto então ele terá um valor diferente de zero no vetor. Assim para atribuir um valor numérico entre um documento e uma consulta, o modelo mede a similaridade entre o vetor de consulta(esta consulta também tem um vetor de termos) e o vetor de documento.
  Geralmente, o ângulo entre dois vetores é utilizado como uma medida de divergência entre os vetores e o cosseno do ângulo é utilizado como o valor de similaridade. O cosseno vale 1.0 para vetores idênticos e 0.0 para vetores ortogonais. O produto interno é usado entre dois vetores como medida de similaridade. Se d é o vetor de documento e q é o vetor de consulta, logo a similaridade entre d e q pode ser representada atraves da equação 2.2.

$$sim_{d,q} = \sum_{t_i \in q,d} w_{t_i,q}.w_{t_i,d} \tag{2.2}$$

  Onde $w_{t_i,q}$ é o valor do i-ésimo componente no vetor de consulta q e $w_{t_i,d}$ e o i-ésimo componente no vetor de documento. Logo pode-se somar sobre os termos comuns na consulta e no documento desde que qualquer termo não presente na consulta tenha $w_{t_i,q}$ como valor 0, ou não presente no documento com $w_{t_i,d}$ igual a 0. O $w_{t_i,q}$ é frequentemente referenciado como peso do termo i no documento d.

- Recuperação Probabilística: Tenta formalizar as ideias por trás da recuperação ordenada em termos de teoria de probabilidades. A probabilidade a ser avaliada é de um documento ser relevante ou não para a consulta.

Capítulo 3

Recuperação de imagens baseada em ontologias para o sistema Arquigrafia

As aplicações web tradicionais de acervo digital de imagens que utilizam a informação das palavras chave ou título das imagens para executar buscas, retornam muitas vezes conteúdo com pouca relevância ou apresentando resultados não satisfatórios.

A utilização de técnicas tradicionais sem nenhum tratamento da informação relacionada com a imagem não é suficiente para obter resultados relevantes para uma busca realizada pelo usuário.

A dificuldade para o motor de busca reside no fato de que o usuário é livre para utilizar a linguagem natural na sua consulta com o fim de encontrar as imagens e informações por ele requeridas. Neste trabalho optou-se por explorar uma abordagem para melhorar a busca de informação das imagens, utilizando ontologias de aplicação para adicionar à consulta termos similares aos termos da consulta.

Por exemplo se o usuário busca *edificação* podemos incluir à consulta novos termos que pertencem à ontologia como prédio, construção, etc. Essa abordagem também faz uso dos modelos de recuperação da informação.

Na literatura encontramos alguns trabalhos que também usaram a abordagem de ontologias e modelos de recuperação da informação para problemas similares.

Fernández *et al.* (2011) apresenta quatro diferentes abordagens identificadas no estado da arte, de acordo com seu grau de formalidade e complexidade de uso. No primeiro nível, as consultas são expressas pelo significado de palavras-chave, é a forma tradicional de consulta, porém é pouco expressivo, já que a informação precisa estar representada como um conjunto de termos sem qualquer relação explícita entre eles. O segundo nível envolve uma representação em linguagem natural. Este tipo de consulta fornece mais informação que uma abordagem por palavras-chave, já que uma análise linguística pode ser realizada para extrair informação sintática, tais como sujeito, predicado, objeto e outros detalhes da sentença. O terceiro nível em formalidade é retratado por um sistema de linguagem natural controlada, na qual as consultas podem ser expressas adicionando etiquetas que representam propriedades, valores ou objetos dentro da consulta. Este tipo de consulta pode ser facilmente processada e mapeada a suas correspondentes classes, propriedades e valores de um esquema ou ontologia descrevendo o espaço de busca, facilitando assim a aquisição de informação semanticamente relacionada. No quarto nível temos a maior formalidade em sistemas de busca baseados em ontologias, que utilizam linguagens de consulta para ontologias tais como RDQL, SPARQL entre outros. A potência expressiva plena deste tipo de consulta permite aos sistemas automaticamente recuperar numa forma altamente precisa a informação que satisfaz as necessidades do usuário.

Outro trabalho de Díaz-Galiano *et al.* (2009) propõe o uso da ontologia médica MeSH com a técnica de expansão de consultas para incluir termos médicos relacionados à consulta do usuário.

Trillo (2005) propôs a abordagem utilizando uma ontologia e técnicas de recuperação de informação, para melhorar a consulta do usuário e recuperar vídeos baseados nas suas transcrições, conseguindo

com esta abordagem recuperar os vídeos relacionados à busca feita pelo usuário.

3.1 Visão geral da abordagem proposta

A metodologia proposta é estruturada em quatro fases (figura 3.1).
A primeira fase consiste na construção da ontologia do domínio de arquitetura e orientada ao sistema de compartilhamento de imagens do sistema Arquigrafia.
A segunda fase é a fase de indexação e pré-processamento na qual preparamos a consulta dada pelo usuário em linguagem natural, corrigindo erros de escrita e colocando pesos para cada termo com o fim de gerar a consulta para a próxima fase.
A terceira fase, consiste na expansão da consulta, que utiliza a ontologia e o índice invertido, com o objetivo de melhorar a consulta original e assim enviá-la para a última fase que é a recuperação das imagens.

O fluxo do processo desta abordagem pode ser observado na figura 3.1.

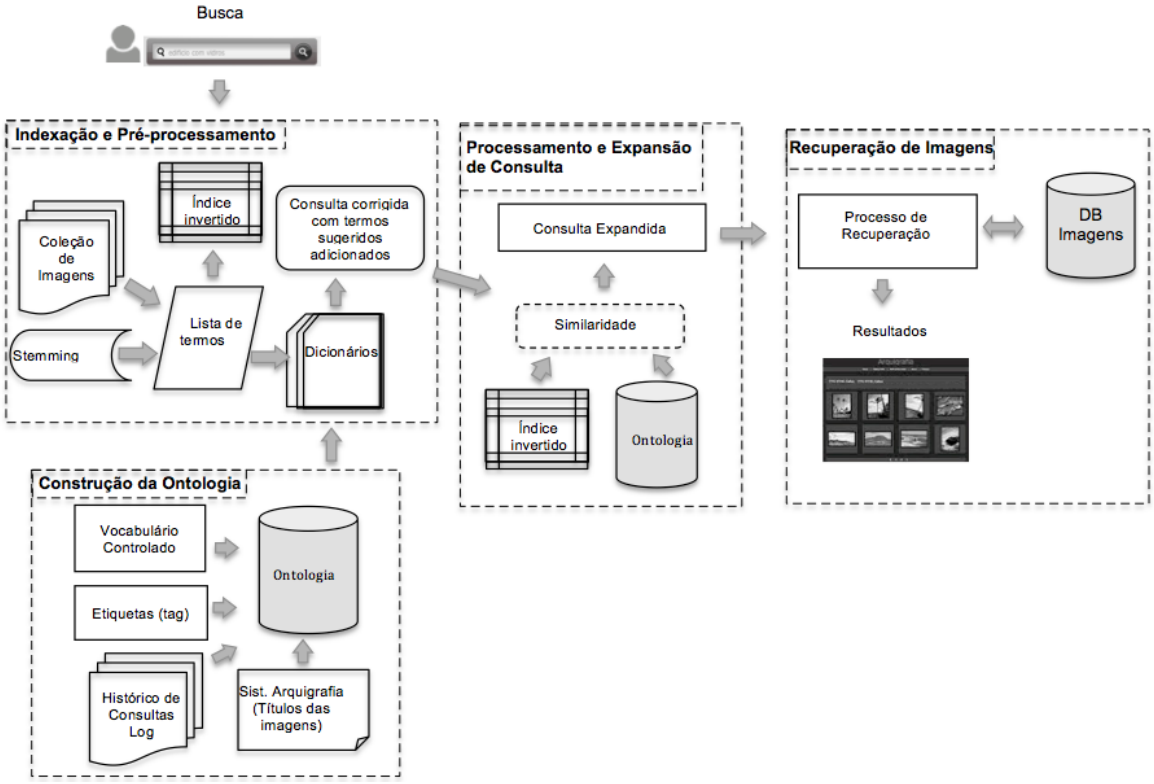


Figura 3.1: *As quatro fases que são parte de nossa abordagem.*

3.2 Fase da construção da ontologia

Como as ontologias provêem uma forma estruturada para descrever o conhecimento, é preciso descrever as entidades ou os conceitos, as propriedades e as relações que existem para um domínio específico, e para este trabalho, tratou-se o domínio do sistema Arquigrafia. Para construção desta ontologia, foi utilizada a metodologia sugerida por Guizzardi (2000).

3.2.1 Propósito e especificação de requisitos

O propósito da construção da ontologia é melhorar as consultas dadas pelo usuário no sistema Arquigrafia, permitindo adicionar termos relacionados com a ontologia, para dessa maneira obter

uma consulta expandida que permita obter melhores resultados para a busca.

Para isso a ontologia trabalha com um conjunto de informação que cobre parte da área de arquitetura, As questões de competência que determinaram este escopo foram extraídas da análise de um conjunto de informações acessíveis que contêm o vocabulário controlado para a área de arquitetura, o histórico de consultas, as etiquetas, e a informação disponível no sistema Arquigrafia.
O conjunto de informações mencionado ajudou a obter as perguntas de competência para definir o ambiente da ontologia, e a maior parte dessas perguntas seguem um padrão encontrado no histórico de consultas.

Algumas questões de competência que ajudaram a definir o escopo da ontologia foram:

- Quem é o autor do edifício de Saúde "Sul América"?
- Quais são os nomes dos autores dos edifícios de Saúde?
- Quais são os edifícios de transporte que utilizaram o "ferro"como material?
- Quais são os museus localizados na cidade de São Paulo?
- Quais foram os materiais utilizados para o museu "Theo Brandão"?
- Quem é o autor da imagem do hospital "Sul América"?

3.2.2 Captura da ontologia

Para a captura da ontologia utilizou-se a informação relevante de Arquigrafia como título da imagem, descrição da imagem, autor da imagem, localização da edificação da imagem e as etiquetas relacionadas às imagens que foram criadas pelos usuários, também foi utilizado o vocabulário controlado e o histórico das consultas do sistema Arquigrafia, todo esse conjunto de informação ajudou na criação dos conceitos, das propriedades, das relações e das instâncias que são parte da ontologia.
A construção da ontologia foi dividida em quatro etapas:

- A primeira etapa é a construção de hierarquia das classes e dos conceitos da ontologia relacionados ao domínio da arquitetura, para isto utilizou-se o vocabulário controlado da USP (que tem uma lista de termos relacionadas ao campo da arquitetura) e a informação disponível no sistema Arquigrafia, como títulos e etiquetas das imagens registradas pelos usuários no sistema.

  Na figura 3.2, é apresentada uma hierarquia de classes para o domínio de arquitetura e orientado para o sistema Arquigrafia. Nesta hierarquia existe a relação "is a", a qual é utilizada entre classes, assim cada filho de uma classe é uma subclasse, e está relacionado uma classe pai através da relação "is a", como exemplo temos que a classe *AutorDaObra* "is a"*Autor*, o qual significa que cada instância ou indivíduo de *AutorDaObra* é uma instância de *Autor*. As classes hierárquicas foram construídas baseadas na análise dos documentos mencionados.

  A figura 3.3 mostra um exemplo das subclasses criadas para a classe *Projeto de Arquitetura*.

- A segunda etapa é a criação das propriedades de objeto e tipo de dados na ontologia, para o qual se utilizou o histórico de consultas do sistema Arquigrafia.

  As propriedades de objeto pertencentes à ontologia:

  - EhParte
  - PertenceAColecao
  - SaoObservacoes
  - TemObservacoes
  - TemAutorDaObra
  - EhAutorDaObra

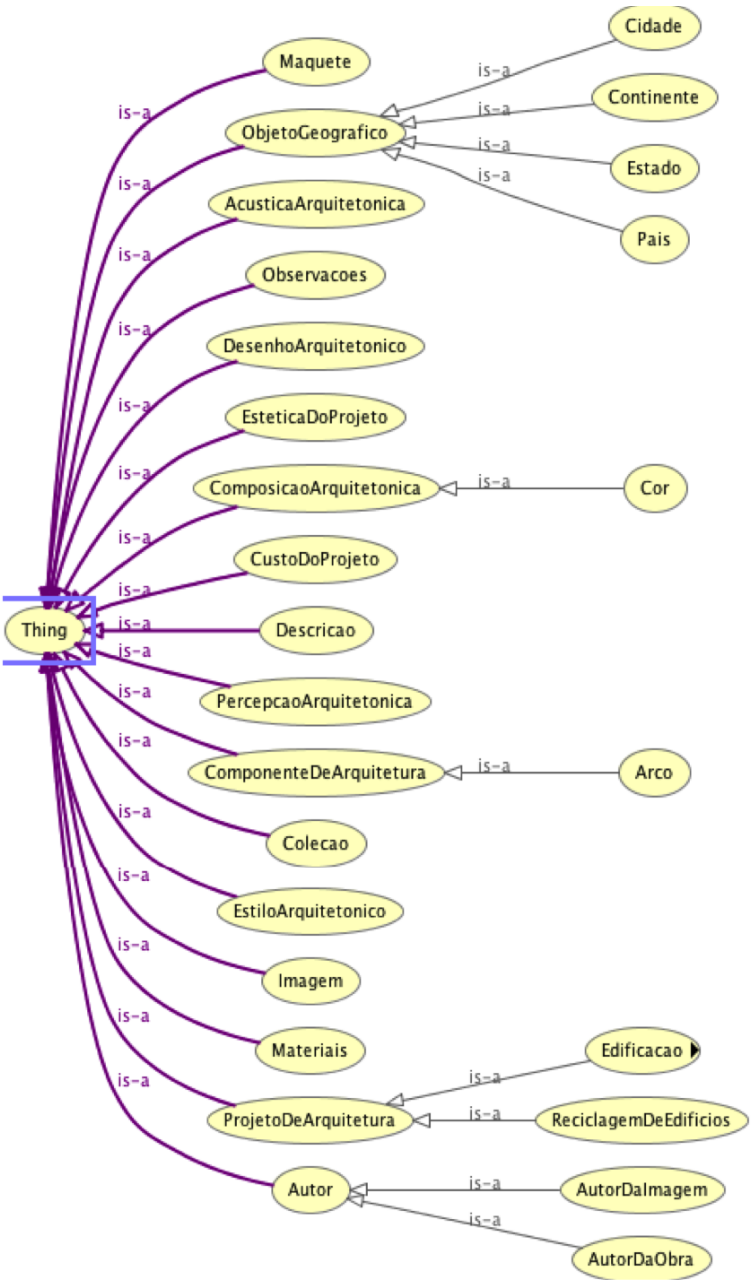


Figura 3.2: *Hierarquia de classes da ontologia no domínio de arquitetura voltado para Arquigrafia.*

- TemAutorDaImagem
- EhAutorDaImagem
- TemDescricao
- EhDescricao
- TemImagem
- EhImagem
- TemComponente
- EhComponente
- TemComposicao
- EhComposicao
- UsaMaterial
- EhMaterial
- LocalizadoEm
- EhLocalizado

As propriedades de Tipo de Dados:

  – TemDataImagem
  – TemDataDaObra

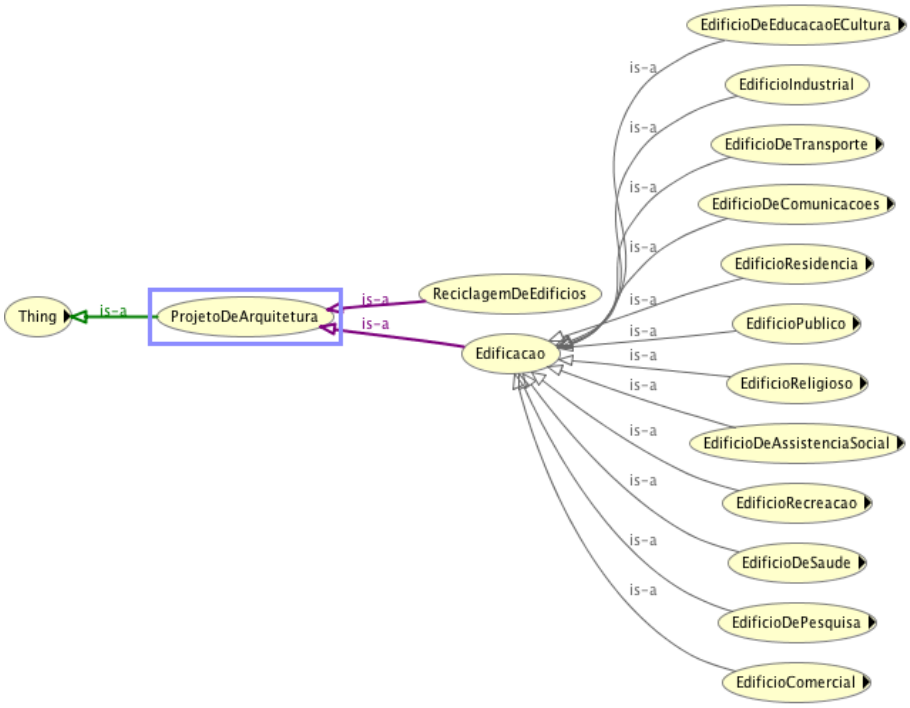


Figura 3.3: *Hierarquia da Subclasse ProjetoArquitetura.*

– TemDataUpload

- A terceira etapa consiste na criação de relacionamentos das classes e para isso também utilizou-se o histórico de consultas.
  A seguir vemos alguns relacionamentos da ontologia no domínio de arquitetura para o sistema de compartilhamento de imagens.

  Por exemplo, temos um fragmento na sintaxe OWL2 para a classe *EdificioDeEducacaoECultura*

```
- Edificacao "hassubclass" EdificioDeEducacaoECultura
<SubClassOf>
         <Class IRI="#EdificioDeEducacaoECultura"/>
    <Class IRI="#Edificacao"/>
</SubClassOf>
```

  A relação "hassubclass"que existe entre a classe *Edificacao* e a classe *EdificioDeEducacaoECultura* significa que a classe *Edificacao* é uma superclasse ou pai da classe *EdificioDeEducacaoECultura* e que todas as instâncias e indivíduos da classe *EdificioDeEducacaoECultura* também serão instâncias da classe *Edificacao*.
  Na sintaxe OWL 2 a relação "hassubclass"é representada como <SubClassOf>.

  Outro exemplo com a relação "SubClassOf"que relaciona à classe *Museu* com a classe *EdificioDeEducacaoECultura* dado na sintaxe OWL 2.

```
- Museu "SubClassOf" EdificioDeEducacaoECultura
<SubClassOf>
<Class IRI="#Museu"/>
<Class IRI="#EdificioDeEducacaoECultura"/>
</SubClassOf>
```

A relação "SubClassOf"que existe entre a classe *Museu* e a classe *EdificioDeEducacaoECultura* quer dizer que todos os indivíduos da classe *Museu* são indivíduos da classe *EdificioDeEducacaoECultura* e também são indivíduos da classe *Edificacao*, posto que *EdificioDeEducacaoECultura* é subclasse de *Edificacao*.

Um exemplo para a relação "hasindividual"em sintaxe OWL 2:

```
- Museu "hasindividual" MuseuJudaico.
<ClassAssertion>
<Class IRI="#Museu"/>
<NamedIndividual IRI="#MuseuJudaico"/>
</ClassAssertion>
```

A relação "hasindividual"que existe entre a classe *Museu* e a classe *EdificioDeEducacaoECultura* significa que a classe *Museu* tem uma instância denominada *MuseuJudaico* ou que *MuseuJudaico* é de tipo *Museu*.

Exemplo para uma relação de tipo de dado "TemDataUpload":

```
- IMuseuJudaico2236 "TemDataUpload" "08/04/2013"^^dataTime.
<DataPropertyAssertion>
        <DataProperty IRI="#TemDataUpload"/>
        <NamedIndividual IRI="#IMuseuJudaico2236"/>
        <Literal datatypeIRI="&xsd;dateTime">2013-04-08T08:08:03</Literal>
     </DataPropertyAssertion>
```

A relação existente entre o indivíduo *IMuseuJudaico2236* da classe *Imagem* e o tipo de dado <"08/04/2013"^^dataTime> é chamada propriedade dado presente no OWL 2, este tipo de propriedade relaciona a uma instância com um literal, que para este exemplo é *dataTime*, quer dizer que para utilizar a relação *TemDataUpload* é preciso ter o formato de data.

Uma relação de propriedade de objeto criada em nossa ontologia para a instância *IMuseuJudaico2236* é:

```
- IMuseuJudaico2236 "TemAutorDaImagem"  GustavoAntonioDeOliveira.
<ObjectPropertyAssertion>
         <ObjectProperty IRI="#TemAutorDaImagem"/>
         <NamedIndividual IRI="#IMuseuJudaico2236"/>
         <NamedIndividual IRI="#GustavoAntonioDeOliveira"/>
     </ObjectPropertyAssertion>
```

Podemos ver que a propriedade de objeto "TemAutorDaImagem"relaciona as instâncias *IMuseuJudaico2236* e *GustavoAntonioDeOliveira* das classes *Imagem* e *AutorDaImagem* respectivamente, isto pode ser observado claramente na sintaxe OWL 2 acima descrita.

Outra relação de propriedade de objeto criada em nossa ontologia para a instância *MuseuJudaico*, é a propriedade de objeto "LocalizadoEm"que relaciona às instâncias da classe *Museu* com a instância da classe *Cidade* as quais são *MuseuJudaico* e *CidadeSaoPaulo* respectivamente.

```
- MuseuJudaico "LocalizadoEm" CidadeSaoPaulo.
<ObjectPropertyAssertion>
<ObjectProperty IRI="#LocalizadoEm"/>
<NamedIndividual IRI="#MuseuJudaico"/>
<NamedIndividual IRI="#CidadeSaoPaulo"/>
</ObjectPropertyAssertion>
```

- A quarta e última etapa desta fase foi a criação de instâncias para cada classe e subclasse, baseado na informação obtida do sistema Arquigrafia disponível na web, nas etiquetas relacionadas a cada imagem.

  Por exemplo as instâncias criadas para a classe *Museu* foram obtidas da informação disponível na web de Arquigrafia e das etiquetas.

  ```
  - Museu Judaico.
  - Museu Paulista.
  - Museu Theo Brandao.
  ```

  Outro exemplo de instâncias obtidas da lista de etiquetas pertencente a Arquigrafia para a classe *Material*.

  ```
  - Madeira
  - Pedra
  - Vidro
  - Gesso
  - Ferro
  - Cimento
  - Cerâmica
  ```

3.2.3 Formalização da ontologia

Utilizou-se a linguagem OWL 2 com o editor Protégé para a construção de nossa ontologia para o Arquigrafia.

- Para a criação de classes o Protégé possui uma aba denominada "Classes"na qual pode-se criar uma hierarquia de classes.
  Na figura 3.4 temos as classes da ontologia proposta criadas no Protégé.

- Para criação de propriedades de objeto o Protégé possui uma aba denominada "Object Properties"na qual são inseridas informações utilizadas para relacionar duas instâncias de distintas classes, além disso permite descrever à propriedade através de suas características descritas na seção 2.
  Por exemplo a propriedade *TemMaterial* tem a característica de ser assimétrica e irreflexiva.
  Na figura 3.5 temos as propriedades de objetos criadas para nossa ontologia.

- Para criação de propriedades de dados o Protégé tem uma aba denominada "Data Properties"na qual pode-se inserir as relações que são utilizadas para relacionar uma instância de uma classe escolhida com um literal ou tipo de dado.
  Por exemplo a propriedade *TemDataDaObra* tem como tipo de dado *String*.
  Na figura 3.6 temos as propriedades de dados criados para nossa ontologia.

- Para a criação de indivíduos, o Protégé também tem uma aba denominada "Individuals"na qual pode-se inserir instâncias pertencentes a cada classe.
  Por exemplo a instância *cerâmica* foi criada para pertencer à classe *Materiais*.
  Na figura 3.7 temos os indivíduos criados para a classe *Materiais*.

- Para a criação das relações já seja para utilizar as propriedades de objetos ou as propriedades de dados, devemos ir à aba "Individual"do Protégé e escolher a classe e a instância com a qual trabalharemos, para logo escolher a propriedade de objeto a relacionar com outra instância e escolher a propriedade de dados a relacionar com o tipo de dado necessário.

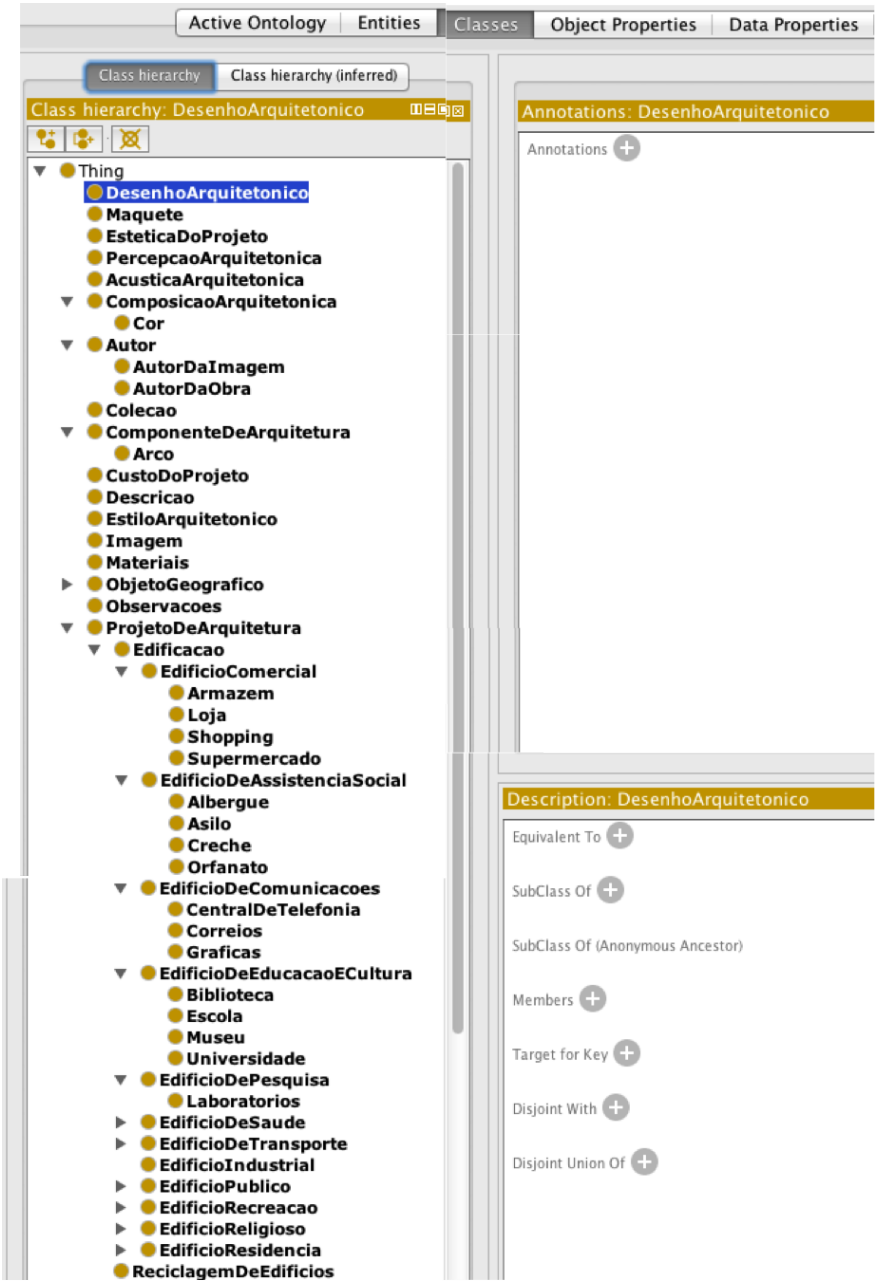


Figura 3.4: *Classes criadas no Protégé.*

Por exemplo na classe *Imagem*, temos o indivíduo *IMuseuJudaico2236* que tem como propriedade de objeto *"TemAutorDaImagem"* que está relacionado à instância *GustavoAntonioDeOliveira* da classe *AutorDaImagem*.
Além disso, o indivíduo *IMuseuJudaico2236* tem a propriedade de dado *"TemDataUpload"* relacionada ao tipo de dado *dateTime* com o valor *2013-04-08T08:08:03*.
A figura 3.8 mostra o exemplo acima descrito.

3.2.4 Integração com ontologias existentes

Para a construção da ontologia proposta neste trabalho não reutilizamos outras ontologias existentes, porque a ontologia está baseada no domínio de arquitetura porém voltado ao sistema Arquigrafia e nós só consideramos para esta primeira iteração, utilizar a informação relevante existente nos títulos, nas descrições e nas etiquetas relacionadas às imagens, e na informação presente no histórico de consultas de Arquigrafia. Por esse motivo foi necessário a construção completa da ontologia, mas

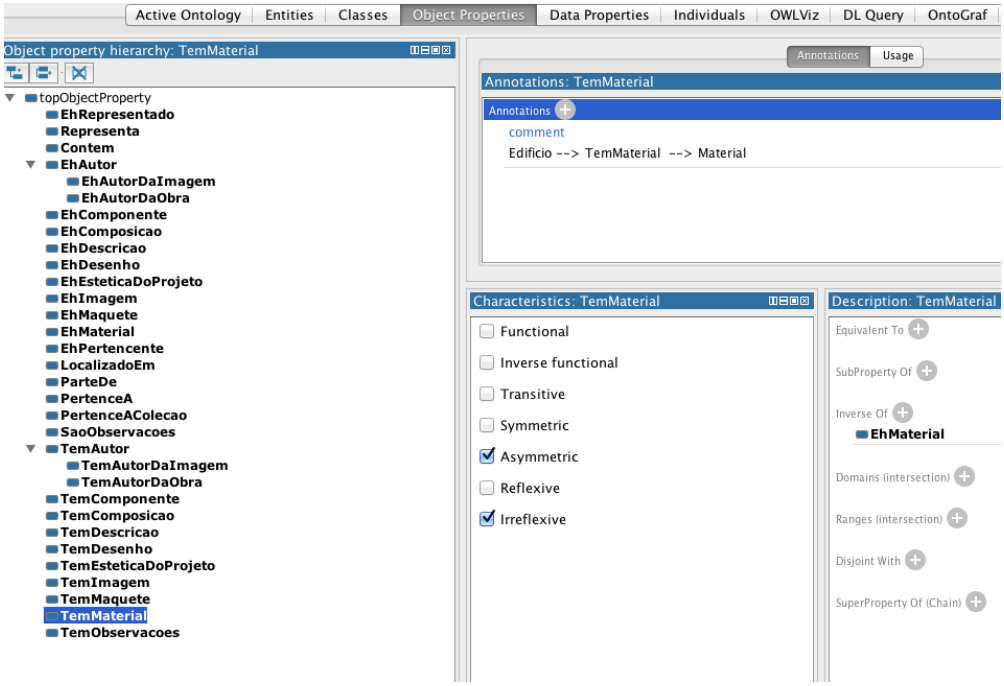


Figura 3.5: *Lista das propriedades de objeto criadas no Protégé para nossa ontologia.*

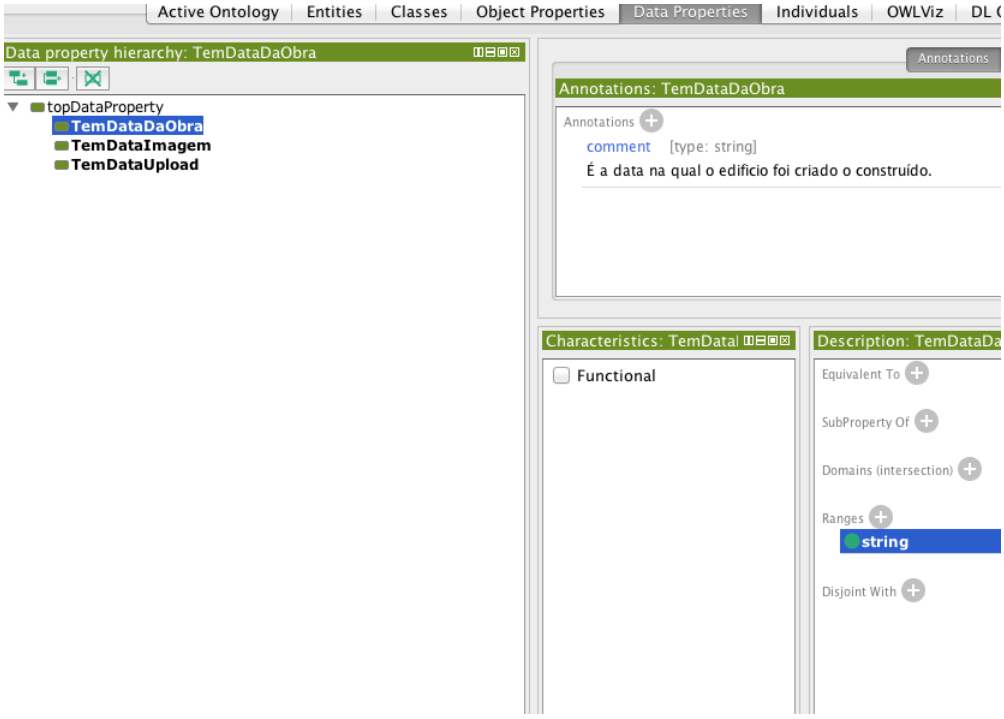


Figura 3.6: *Lista das propriedades de dados criadas no Protégé para nossa ontologia.*

para enriquecer a ontologia, no futuro pensamos integrá-la com outras ontologias relacionadas à área de arquitetura, como por exemplo temos o trabalho de Liu *et al.* (2006) "Ontology based semantic modeling for chinese ancient architectures"e do Hois *et al.* (2009) "Modular Ontologies for Architectural Design".

3.2.5 Avaliação e documentação

Para saber se a ontologia está construída corretamente a mesma deve responder as questões formuladas na etapa de propósito e especificação de requisitos, para isto criamos consultas utilizando

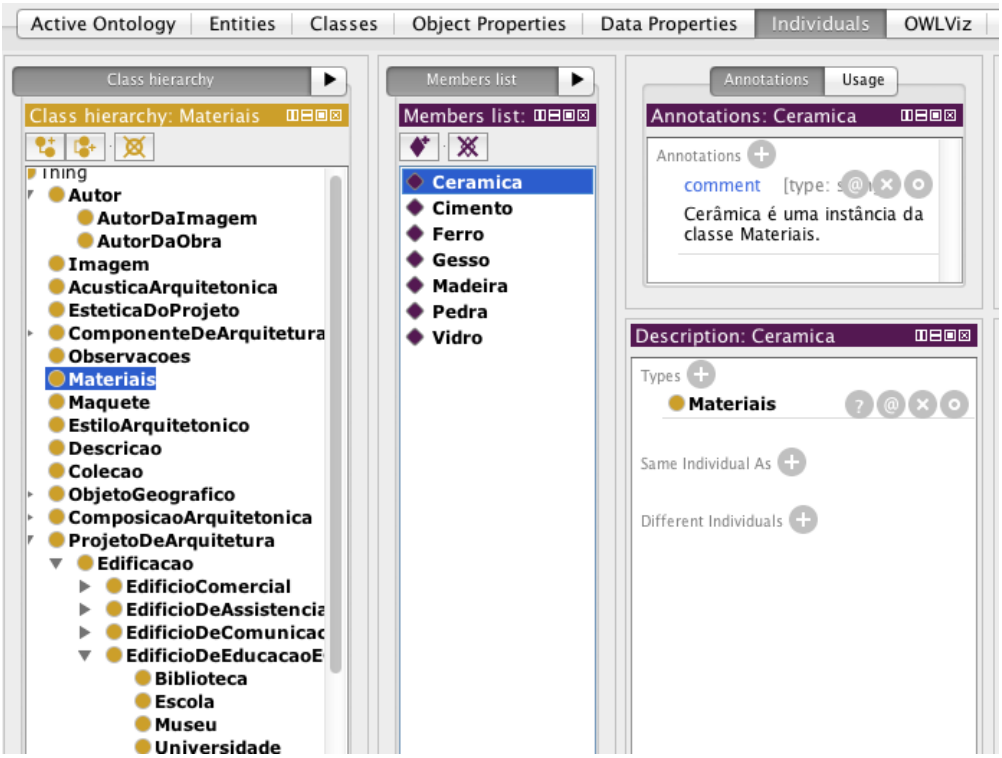


Figura 3.7: *Lista dos indivíduos criados para a classe Materiais no Protégé para nossa ontologia.*

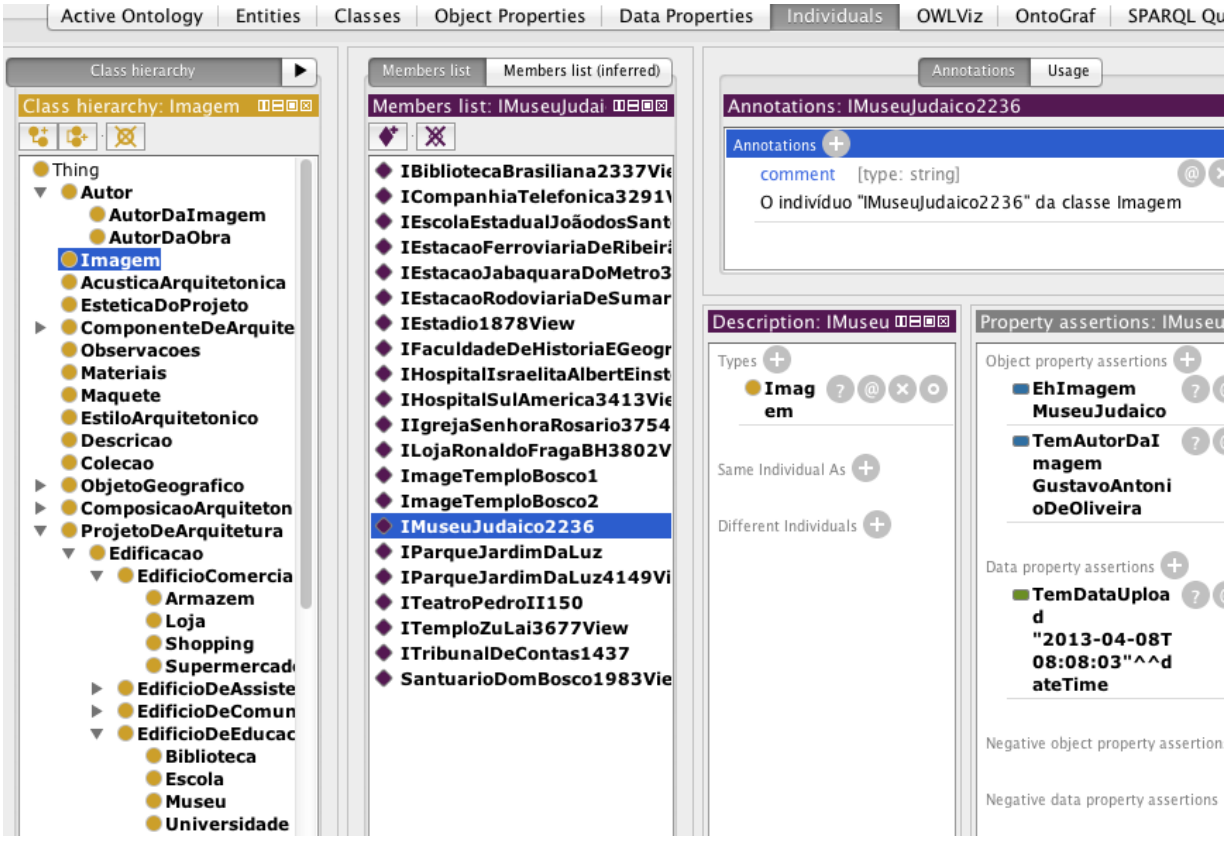


Figura 3.8: *Propriedades de objeto e de dados relacionados à instância IMuseuJudaico2236 da clase Imagem no Protégé para nossa ontologia.*

a linguagem SPARQL[1].

[1]SPARQL: http://sparql.org/

Os resultados destas consultas deverão ser avaliadas por um especialista na área de arquitetura. Por exemplo, temos a seguinte consulta em SPARQL que responde a questão "Quais são os materiais utilizados na construção dos edifícios de transporte?", o resultado desta consulta é mostrado na figura 3.9.

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p: <http://www.semanticweb.org/ontologies/2014/5/untitled-ontology-20#>

SELECT ?edificio  ?material
WHERE {
  ?edificio rdf:type ?type.
  ?type rdfs:subClassOf* p:EdificioDeTransporte.
  ?edificio p:TemMaterial ?material.
}
```

edificio	material
EstacaoRodoviariaDeSumare	Ferro
EstacaoJabaquaraDoMetro	Pedra
EstacaoJabaquaraDoMetro	Vidro
EstacaoJabaquaraDoMetro	Ferro

Figura 3.9: *Exemplo de uma consulta "Quais são os materiais utilizados na construção dos edifícios de transporte?"realizada em SPARQL.*

3.3 Fase de Indexação e Pré-processamento

Para o processo de indexação vamos considerar o seguinte:

a) Primeiro, criamos a coleção de imagens utilizando a informação do sistema de compartilhamento de imagens Arquigrafia. Nesta coleção temos documentos com conteúdo relevante de cada imagem. Nós consideramos como conteúdo relevante os termos presentes no título da cada imagem, na descrição de cada imagem, e nas etiquetas relacionadas às imagens, que serão parte de cada documento.

b) Segundo, criamos uma lista de termos, que é extraída manualmente da ontologia para a recuperação de nomes das classes, instâncias e propriedades, também nesta lista adicionamos os termos extraídos dos documentos das imagens. Logo aplicamos a remoção de afixos na lista, para obter uma lista nova só com as raízes das palavras a qual é menor em comprimento.

c) Terceiro, criamos o índice invertido utilizando a lista de termos, desconsiderando as palavras de parada. No arquivo de índice invertido teremos a ligação de cada termo para os documentos onde ele ocorre, com informação do número de ocorrências do termo no documento, a frequência dos termos. Também consideraremos o peso inicial com valor zero para cada termo.

Para a parte do pré-processamento:
Nesta fase temos a consulta em linguagem natural escrita pelo usuário, para o qual aplicamos:

- A correção de cada palavra pertencente à consulta que tenha erro ortográfico. Para corrigir estes erros utilizamos a API Jazzy, a qual precisa de um dicionário de palavras em português e um dicionário específico que terá as palavras chaves, ou seja, palavras relevantes pertencentes à informação do Arquigrafia e os termos pertencentes à ontologia. Este processo permitirá obter sugestões de palavras corretas para cada palavra errada. As palavras sugeridas formarão parte da consulta, ou seja, serão adicionadas à consulta do usuário.

- Ao conjunto de palavras pertencentes à consulta incluídas as palavras sugeridas, aplicamos a remoção de afixos, para obter só as raízes das palavras.
- A lista de termos pertencentes à coleção de imagens é usada para criar o vetor de consulta, e para cada termo colocamos o valor 1 se o termo está presente na consulta, caso contrario utilizamos o valor zero. Também adicionamos um valor extra se o termo presente na consulta está presente na ontologia, além disso se o termo não é comum também se somará um valor adicional.

3.4 Fase de processamento e consulta expandida

A expansão de consultas está baseada na ontologia, para esta expansão analisamos os conceitos e os relacionamentos de um domínio específico com o fim de adicionar os termos derivados dessa análise, para ser parte da consulta que é dada da fase anterior.

Nesta fase alteramos os pesos dos termos da consulta sendo necessário utilizar o arquivo de índice invertido e comparar a similaridade existente entre cada termo da consulta com os termos do índice. Se o termo está presente no índice invertido o peso dele é atualizado. Logo deve-se escolher os termos que obtiveram os pesos máximos dos termos similares.

Para encontrar a similaridade de cada termo t da consulta q utilizamos a equação 3.1 dada na definição do modelo de espaço vetorial na seção 2.9.2 temos:

$$sim(t,q) = \sum_{t_i \in q} w_{i,q} \times sim(t,t_i) \tag{3.1}$$

onde $w_{i,q}$ é o peso do termo t_i pertencente à consulta q e a $sim(t,t_i)$ é o cálculo de similaridade entre todos os termos t do índice invertido com os termos t_i da consulta. Este cálculo é importante porque analisa se os termos t e t_i são iguais ou similares.

Para escolher o termo t do índice invertido a ser adicionado à consulta, é importante atualizar seus pesos fazendo uso da similaridade acima mencionada e aplicando a equação 3.2

$$w_{ex}(t,q) = \frac{sim(t,q)}{\sum_{t_i \in q} w_{i,q}} \tag{3.2}$$

A seguir, deve-se aplicar o algoritmo de expansão de consulta ("Expansão_Consulta") feito pelo autor Trillo (2005) que utiliza o método $SimTermos$.

O método $SimTermos$ tem um algoritmo que incorpora os conceitos ou classes associadas ao termo, baseados nos termos t do índice e do termo t da consulta, para poder aplicá-la na equação 3.3, a qual analisará os termos e retornará 1 se os termos são sinônimos, e retornará 0 se os termos analisados não são similares, ou seja, não existe algum termo sinônimo que esteja presente na ontologia.

A equação 3.3, utiliza dois fatores de similaridade sobre a hierarquia dos conceitos e relacionamentos pertencentes a nossa ontologia.

$$sim(t_1,t_2) = \beta \times \frac{2 \times \log P(sup)}{\log P(t_1) + \log P(t_2)} + (1-\beta) \times \frac{prop(t_1,t_2) + prop(t_2,t_1)}{prop(t_1) + prop(t_2)} \tag{3.3}$$

O primeiro fator foi proposto por Lin (1998) que considera que sup é a classe superior e $P(sup)$ é a frequência total das classes que possuem os termos t_1 e t_2 como sub-classes da ontologia, as $P(t_1)$ e $P(t_2)$ que são as frequências do termo t_1 e o termo t_2 respectivamente.

O segundo fator calcula o relacionamento dos termos t_1 e t_2 utilizando a contagem de suas propriedades.

$prop(t_1,t_2)$ representa a quantidade de propriedades que tem t_1 como domínio e t_2 como imagem,

Algoritmo 1 Expansão_Consulta (consulta,indice,ontologia)

Entrada:
consulta: consulta
indice: índice invertido
ontologia: ontologia
Saída:
consulta_expandida: consulta expandida

1: $peso\_consulta \leftarrow 0$
2: **para cada** termo u da *consulta* **faça**
3: $peso\_consulta \leftarrow peso\_consulta + consulta(u).peso$
4: **fim**
5: **para cada** termo t de *indice* **faça**
6: $sim(t) \leftarrow 0$
7: **para cada** termo u da *consulta* **faça**
8: $sim(t) \leftarrow sim(t) + consulta(u).peso * SimTermos(t, u, ontologia, indice)$
9: **fim**
10: $w_{ex}(t).peso \leftarrow sim(t)/peso\_consulta$
11: **fim**
12: **para cada** termo u da *consulta* **faça**
13: **se** $w_{ex}(t)$ é um dos max_{exp} termos com maior peso **então**
14: $consulta\_expandida(t) \leftarrow consulta(t)$
15: $consulta\_expandida(t).peso \leftarrow consulta\_expandida(t).peso + w_{ex}(t).peso$
16: **fim**
17: **fim**

de modo inverso acontece com $prop(t_2, t_1)$, logo a $prop(t_1)$ e $prop(t_2)$ que armazenam a quantidade de propriedades dos termos t_1 e t_2 respectivamente.

A seguir temos o algoritmo de similaridade de termos ("SimTermos") feito pelo autor Trillo (2005) utiliza o método $SimTermos$.

3.5 Fase de recuperação das imagens

Na etapa para recuperar as imagens relevantes, utilizaremos a informação relevante relacionado a cada imagem que chamaremos de documento da imagem e vamos a representá-lo num vetor espacial que irá conter os pesos de todos os termos pertencentes à imagem. Também vamos ter outro vetor espacial que contém os pesos dos termos da consulta. Estes vetores pertencem à teoria do modelo de espaço vetorial descrito na seção 2.9.2 a qual tem como base o peso dos termos e para calcular a similaridade existente entre o documento da imagem e a consulta, que permitirá obter uma lista de imagens em ordem de relevância.

Para este cálculo utilizaremos as equações 3.4 e 3.5 as quais serão utilizados no algoritmo de Recuperação ("Recupera") feito pelo autor Trillo (2005).

Primeiro aplicaremos a técnica do cosseno do ângulo formado pelo vetor de consulta e o vetor de documento de imagem como podemos observar na equação 3.4:

$$cos_{doci,q} = \sum_{t_i \in doci} w_{i,q} \times w_{i,doci} \tag{3.4}$$

Onde $w_{i,q}$ é o peso do termo i na consulta q; e $w_{i,doci}$ é o peso de cada termo i pertencente ao documento da imagem $doci$.

Algoritmo 2 SimTermos(termo1,termo2,ontologia,indice)

Entrada:

$termo1$: primeiro termo
$termo2$: segundo termo
$ontologia$: ontologia
$indice$: indice

Saída:

$similaridade$: valor numérico da similaridade entre os termos

1: **se** $termo1 == termo2$ **então**
2: **retorna** $similaridade \leftarrow 1$
3: **fim**
4: $classes1 \leftarrow onto.pegaClasses(termo1)$
5: $classes2 \leftarrow onto.pegaClasses(termo2)$
6: **se** classes1 é vazio **ou** classes2 é vazio **então**
7: **retorna** $similaridade \leftarrow 0$
8: **fim**
9: **se** classes1 **e** classes2 não é vazio **então**
10: **retorna** $similaridade \leftarrow 1$
11: **fim**
12: $classesSup \leftarrow ontologia.pegaClassesSuperioresComuns(classes1, classes2)$
13: **se** classesSup não é vazio **então**
14: $probTermo1 \leftarrow \frac{indice.pegaFrequencia(classes1)}{indice.totalDocs}$
15: $probTermo2 \leftarrow \frac{indice.pegaFrequencia(classes2)}{indice.totalDocs}$
16: $probTermoSup \leftarrow \frac{indice.pegaFrequencia(classesSup)}{indice.totalDocs}$
17: **se** $\log(probTermo1 \times probTermo2)! = 0$ **então**
18: $atermo \leftarrow \frac{2 \times \log(probTermoSup)}{\log(probTermo1 \times probTermo2)}$
19: **fim**
20: **fim**
21: $(totalProp1, propRelacionadas1) \leftarrow ontologia.contarProp(classes1, classes2)$
22: $(totalProp2, propRelacionadas2) \leftarrow ontologia.contarProp(classes2, classes1)$
23: **se** $totalProp1 + totalProp2 > 0$ **então**
24: $btermo \leftarrow \frac{propRelacionadas1 + propRelacionadas2}{totalProp1 + totalProp2}$
25: **fim**
26: **retorna** $similaridade \leftarrow (\beta \times atermo) + ((1 - \beta) \times btermo)$

Para o cálculo de similaridade temos a equação 3.5:

$$sim_{doci,q} = \frac{cos_{doci,q}}{\sqrt{\sum_{t_j \in doci} w_{j,doci} \times \sum_{t_k \in q} w_{k,q}}} \tag{3.5}$$

Onde $w_{j,doci}$ é o peso de cada termo j do documento da imagem $doci$ e $w_{k,q}$ é a soma de todos os pesos dos termos k da consulta q.

Com estes cálculos obteremos uma lista de todas as imagens com as informações relevantes respectivas que tenham maior similaridade com a consulta expandida, com o fim de retornar as imagens mas relevantes ao usuário.

Algoritmo 3 Recupera (consulta,indice)

Entrada:
consulta: consulta
indice: índice invertido
Saída:
docs_analisados: documentos analisados

1: $peso\_total\_consulta \leftarrow 0$
2: **para cada** termo u da *consulta* **faça**
3: $entrada \leftarrow indice.pegaEntradaTermo(u)$
4: **se** *entrada* é nula **então**
5: processar seguinte termo da consulta
6: **fim**
7: $peso\_consulta \leftarrow (consulta(u).peso)^2$
8: $peso\_total\_consulta \leftarrow peso\_total\_consulta + peso\_consulta$
9: **para cada** documento *doc* na *entrada* **faça**
10: $docs\_analisados(doc).cos \leftarrow docs\_analisados(doc).cos + peso\_consulta * entrada(doc).peso$
11: **fim**
12: **fim**
13: **para cada** documento *doc* em *docs_analisados* **faça**
14: $entrada \leftarrow indice.pegaEntradaDoc(doc)$
15: **para cada** termo t da *entrada* **faça**
16: $docs\_analisados(doc).peso\_doc \leftarrow docs\_analisados(doc).peso\_doc + (entrada(t).peso)^2$
17: **fim**
18: $docs\_analisados(doc).sim \leftarrow \frac{docs\_analisados(doc).cos}{docs\_analisados(doc).peso\_doc * peso\_total\_consulta}$
19: **fim**
20: **retorna** *docs_analisados*

Capítulo 4

Resultados preliminares

Neste capítulo apresentaremos inicialmente uma descrição do conjunto de dados de entrada que foram utilizadas neste trabalho. Também apresentaremos alguns resultados preliminares da fase de construção da ontologia seguindo a metodologia descrita no capítulo 3.

Para a construção da ontologia de aplicação foi utilizado o software Protégé [1] que provê uma interface gráfica para a definição da ontologia. Também utilizamos o plugin de SPARQL[2] para fazer consultas sobre a ontologia.

4.1 Conjunto de dados

4.1.1 Fase de construção de ontologias

Para esta fase os arquivos utilizados foram o vocabulário controlado da USP na área de arquitetura, os arquivos de etiquetas criados pelos usuários, o histórico de consultas (logs) e a informação obtida do sistema de compartilhamento de imagens Arquigrafia.

Utilizou-se os seguintes arquivos em cada estágio.

Estágios ou etapas	Arquivos utilizados
Para a primeira etapa que consiste na criação de conceitos como as classes, subclasses quer dizer a taxonomia das classes na ontologia.	Utilizamos o vocabulário controlado da USP da área de arquitetura, as etiquetas criadas pelo usuário e a informação obtida de Arquigrafia que esta disponível na web.
Para a segunda etapa que consiste na criação de propriedades de objeto e propriedades de dados.	Utilizou-se os arquivos de histórico de consulta.
Para a terceira etapa que consiste na criação de relacionamentos entre instâncias de diferentes classes ou entre uma instância e um tipo de dado, que utiliza as propriedades de objeto e dados.	Utilizou-se os arquivos de histórico de consulta.
Para a quarta etapa que consiste na criação de instâncias	Se utilizaram as etiquetas criadas pelo usuário e a informação relevante obtida de Arquigrafia disponível na web.

[1] Protégé: http://Protégé.stanford.edu/
[2] Sparql: http://sparql.org/

4.1.2 Fase de indexação e pre-processamento

Fase de indexação

Documentos	Detalhes
Coleção de imagens.	Esta coleção contem informação relevante como, os títulos, as descrições e as etiquetas relacionadas às imagens, todas foram extraídas manualmente do sistema de compartilhamento de imagens Arquigrafia. Foram extraídas 16 imagens que foram registradas em um banco de dados local, com o objetivo de serem utilizadas pelo índice invertido.
Lista de termos.	Foi extraída dos documentos da coleção das imagens e dos termos pertencentes à ontologia.
Índice Invertido.	Utilizamos a lista de termos mencionada.

Pre-processamento

Descrição	Ferramentas e arquivos utilizados
Para a correção das palavras com erros ortográficos pertencentes à consulta ingressada pelo usuário.	Utilizamos a ferramenta Jazzy [3] e o dicionário em português br.ispell [4].

4.2 Resultados

4.2.1 Ontologia

Após a analise da classificação de termos arquitetônicos encontrados no vocabulário controlado da USP e a informação relacionada a cada imagem de Arquigrafia como o título, a descrição, os autores e as etiquetas, foi possível a captura de classes da ontologia e a criação taxonômica da mesma. Além as instâncias pertencentes a estas classes foram extraídas informações presentes no Arquigrafia. Como exemplo, a figura 4.1 ilustra as classes e subclasses desenvolvidas no software editor de ontologias Protégé.

As classes, subclasses e instâncias pertencentes à ontologia são descritas na tabela 4.1.

As propriedades de objetos e de dados existentes na ontologia foram criadas em relação à analise do histórico de consultas. A figura 4.2 mostra as propriedades existentes na ontologia.

Na tabela 4.2 são descritas as propriedades de objeto e as relações derivadas pela utilização destas propriedades. Na tabela 4.3 descreveremos as propriedades de dados e as relações existentes pela utilização das mesmas.

4.2.2 Consultas Realizadas

As consultas realizadas foram baseadas nas questões de competência para testar a ontologia e utilizou-se a linguagem SPARQL:
A ontologia está disponível no site $http://www.ime.usp.br/\ msolis/ontologies/ontoArquigrafia1.owl$.
Para as consultas em SPARQL utilizamos o prefixo, "PREFIX p: <http://www.ime.usp.br/ msolis/ontologies/ontoArquigrafia1#>"que pertence à ontologia.

- Quem é o autor do edifício de Saúde *Sul América*?

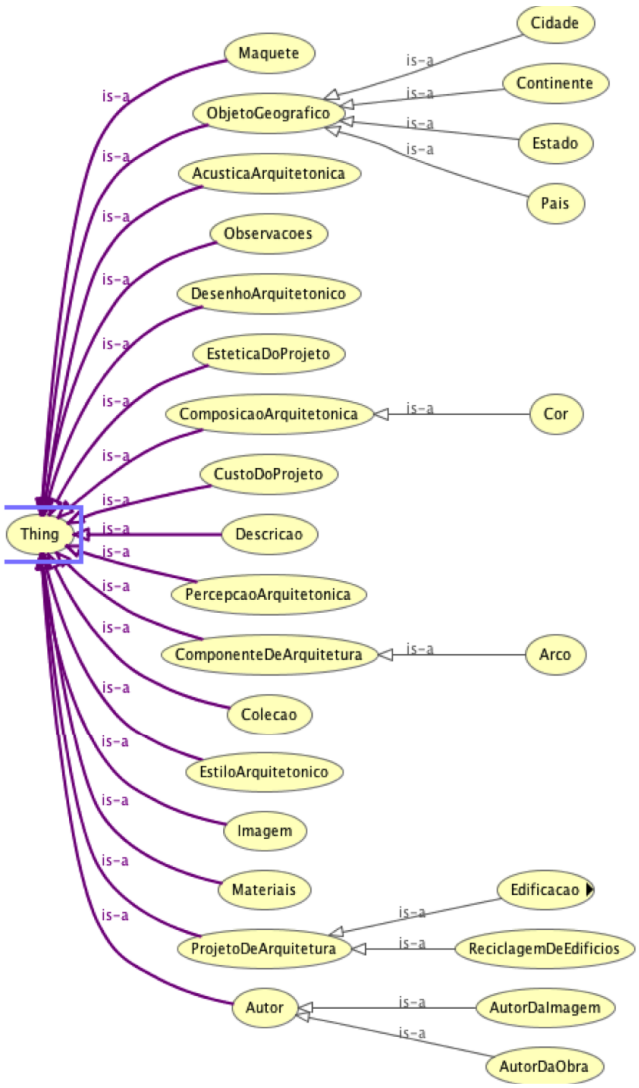


Figura 4.1: *Taxonomia de classes pertencentes à ontologia de domínio arquitetônico voltado para Arquigrafia.*

A figura 4.3 mostra o resultado da consulta.

- Quais são os nomes dos autores dos edifícios de Saúde?

  A figura 4.4 mostra o resultado da consulta.

- Quais são os edifícios de transporte que utilizaram o *ferro* como material?

  A figura 4.5 mostra o resultado da consulta.

- Quais são os museus localizados na cidade de *São Paulo*?

  A figura 4.6 mostra o resultado da consulta.

- Quais foram os materiais utilizados para o museu *Theo Brandao*?

Classes	Comentários e indivíduos
Desenho arquitetônico.	São gráficos produzidos por arquitetos durante o processo de projeto arquitetônico. Algumas instâncias são desenho no computador, desenho no papel.
Maquete	Reprodução tridimensional, em miniatura, de um projeto arquitetônico. Exemplo de instâncias temos maquete virtual e maquete manual a escala
Acústica arquitetônica.	Se importam com o som nos recintos fechados ou semi-abertos e sua transmissão sonora. como exemplo temos os teatros e as igrejas.
Composição Arquitetônica.	Tem os elementos como forma, textura, cor, luz e sombra.
Autor.	Autor tem duas subclasses autor de imagem e autor da obra. Os indivíduos são os nomes dos autores.
Autor Da Imagem.	É quem fez upload da imagem no sistema.
Autor Da Obra.	É o responsável de uma edificação específica.
Coleção.	Esta classe agrupará a um conjunto de imagens. Exemplo de uma instância é a coleção "Quapa".
Componentes de Arquitetura.	Elementos que formam parte da construção da arquitetônica. Exemplo de indivíduos são o concreto, escada,fachada,janela entre outros.
Descrição	Descreve uma imagem pertencente a um edifício.
Estilo arquitetônico	Expressão utilizada na arquitetura no período da historia baseado nas características formais, técnicas e materiais. Exemplo para indivíduos são barroco, neoclássico entre outros.
Materiais	São objetos que foram utilizados pelo projeto de arquitetura. Exemplo para indivíduos são cimento, ferro, gesso, madeira, entre outros.
Objeto Geográfico	É o localização de algum edifício. Tem subclasses como cidade, continente, estado, pais.
Projeto de Arquitetura	Tem subclasses como edificação e reciclagem de edifícios que são os produtos dos processos do projeto de arquitetura.
Edificação	Tem várias subclasses de edifícios como os comerciais, saúde, educação e cultura, comunicações, religioso,recreação, industrial, residência, publico, transporte, assistência social.
Edifício de saúde	tem também subclasses como ambulatório, clinica, consultório, hospital e maternidade. Os indivíduos para estas classes são os nomes de cada edifício.

Tabela 4.1: *Esta tabela mostra as descrições das classes, as subclasses e as instâncias pertencentes à ontologia*

Figura 4.2: *Lista das propriedades de objeto e de dados da ontologia.*

Propriedades objeto	Comentários	Exemplos
Contem	A relação se da entre uma instância da classe **"Continente"** com uma instância da classe. **"Pais"**	América **Contem** Brasil.
EhAutorDaImagem	Relaciona uma instância da classe **"AutorDaImagem"** com uma instância de uma das subclasses da classe **"Edificacao"**.	Hugo MassakiSegawa **EhAutorDaImagem** do Hospital Sul América.
EhAutorDaObra	Relaciona uma instância da classe **"AutorDaObra"** com uma instância da subclasse de **"Edificacao"**.	Oscar Ribeiro de Almeida de Niemeyer **EhAutorDaObra** hospital Sul América".
EhComponente	Relaciona uma instância da classe **"ComponenteDeArquitetura"** com uma instância de alguma subclasse da classe **"Edificacao"**	Janela **EhComponente** companhia telefônica.
EhDescricao	Relaciona uma instância da classe **"Descrição"** com uma instância de alguma subclasse **"Edificacao"**.	Vista geral da companhia telefônica **EhDescricao** companhia telefônica.
EhMaterial	Relaciona uma instância da classe **"Material"** com uma instância de alguma subclasse **"Edificacao"**.	Pedra **EhMaterial** companhia telefônica.
LocalizadoEm	Relaciona uma instância da subclasse de **"Edificacao"** com uma instância da classe **"ObjetoGeografico"**	Hospital Sul América **LocalizadoEm** Rio de Janeiro.
PertenceAColecao	Relaciona uma instância classe **"Imagem"** com uma instância da classe **"Colecao"**.	Imagem do parque Jardim da Luz **PertenceAColecao** Quapa.
TemAutorDaImagem	Relaciona uma instância de alguma subclasse da classe **"Edificacao"** com uma instância da classe **"AutorDaImagem"**.	Hospital Sul América **TemAutorDaImagem** Hugo MassakiSegawa.
TemAutorDaObra	Relaciona uma instância de uma subclasse da classe **"Edificacao"** com uma instância da classe **"AutorDaObra"**.	Biblioteca Brasiliana **TemAutorDaObra** Eduardo de Almeida.
TemComponente	Relaciona uma instância da subclasse da classe **"Edificacao"** com uma instância da classe **"ComponenteDeArquitetura"**.	Estação Ferroviária de Ribeirão preto **TemComponente** concreto.
TemDescricao	Relaciona uma instância da subclasse da classe **"Edificacao"** com uma instância da classe **"Descrição"**.	Estação Jabaquara do Metrô **TemDescricao** Vista geral da estação Jabaquara.
TemMaterial	Relaciona uma instância da subclasses da classe **"Edificacao"** com uma instância da classe **"Material"**.	O estadio Jornalista Mario Filho Maracana **TemMaterial** ferro.

Tabela 4.2: *Esta tabela mostra as descrições das propriedades de objeto utilizadas na ontologia.*

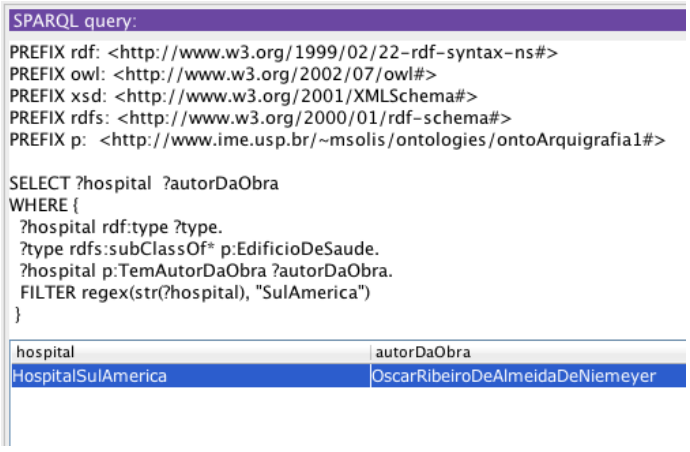


Figura 4.3: *Resultado da consulta* ***"Quem é o autor do edifício de Saúde Sul América?"***

A figura 4.7 mostra o resultado da consulta.

Propriedades de dados	Comentários	Exemplos
TemDataDaObra	Esta propriedade é utilizado para registrar a data na qual o edifício foi construído e a relação existe entre uma instância de algumas das subclasses da classe **"Edificacao"** com o tipo de dado **"String"**	Templo Zu Lai **TemDataDaObra** 1 de outubro de 2003 ∧∧ string.
TemDataImagem	Esta propriedade é utilizado para registrar a data na qual foi tomada a imagem e a relação existe entre uma instância da classe **"Imagem"** com o tipo de dado **"String"**.	Imagem de parque Jardim da Luz **TemDataImagem** Agosto de 2003∧∧string.
TemDataUpload	Esta propriedade é utilizado para registrar a data na qual se fez upload da imagem e a relação se da entre uma instância da classe **"Imagem"** com o tipo de dado **"dateTime"**.	Templo Zu Lai **TemDataUpload** 2013-11-19T08:08:03∧∧dateTime.

Tabela 4.3: *Esta tabela mostra as descrições das propriedades de dados utilizadas na ontologia.*

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p:  <http://www.ime.usp.br/~msolis/ontologies/ontoArquigrafia1#>

SELECT ?edificio  ?autorDaObra
WHERE {
  ?edificio rdf:type ?type.
  ?type rdfs:subClassOf* p:EdificioDeSaude.
  ?edificio p:TemAutorDaObra ?autorDaObra.
}
```

edificio	autorDaObra
HospitalIsraelitaAlbertEinstein	RinoLevi
HospitalSulAmerica	OscarRibeiroDeAlmeidaDeNiemeyer

Figura 4.4: *Resultado da consulta* ***"Quais são os nomes dos autores dos edifícios de Saúde?"***

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p:  <http://www.ime.usp.br/~msolis/ontologies/ontoArquigrafia1#>

SELECT ?edificio  ?material
WHERE {
  ?edificio rdf:type ?type.
  ?type rdfs:subClassOf* p:EdificioDeTransporte.
  ?edificio p:TemMaterial ?material.
FILTER regex(str(?material), "Ferro")
```

edificio	material
EstacaoJabaquaraDoMetro	Ferro
EstacaoRodoviariaDeSumare	Ferro

Figura 4.5: *Resultado da consulta* ***"Quais são os edifícios de transporte que utilizaram o ferro como material?"***

- Quem é o autor da imagem do hospital *Sul América*?

  A figura 4.8 mostra o resultado da consulta.

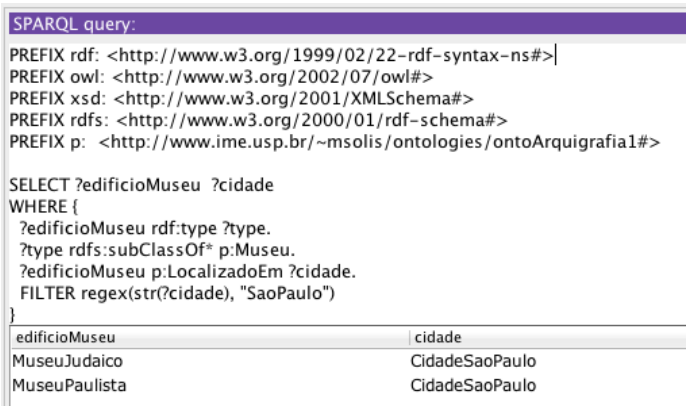


Figura 4.6: *Resultado da consulta* ***"Quais são os museus localizados na cidade de São Paulo?"***

Figura 4.7: *Resultado da consulta* ***"Quais foram os materiais utilizados para o museu Theo Brandao?"***

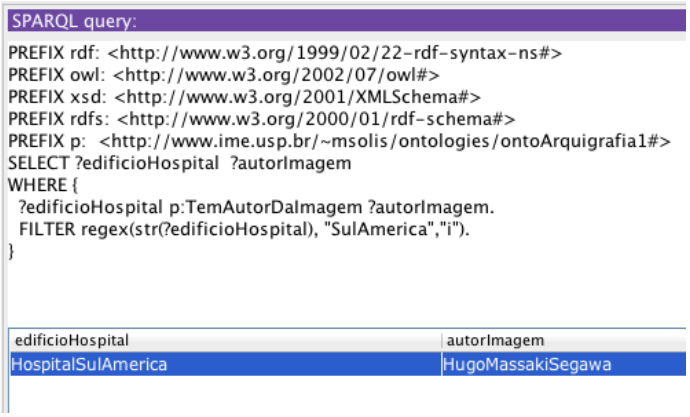


Figura 4.8: *Resultado da consulta* ***"Quem é o autor da imagem do hospital Sul América?"***

4.2.3 Resultado inicial de consultas com e sem ontologias

Nesta seção temos três exemplos de consultas solicitadas pelos usuários; comparando os resultados quando uma ontologia é utilizada e quando não.

a) Primeiro temos um exemplo de inferência. E na figura 4.9 temos informação classificada a respeito da classe edificação.

- Para esta classificação sem a ontologia, se o usuário busca "edificação", obterá como

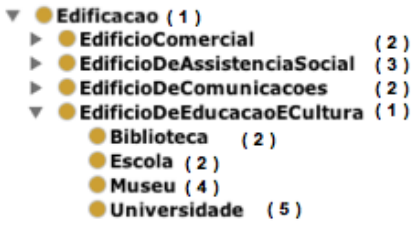


Figura 4.9: *Informação classificada para a classe "Edificacao"*

resultado uma instância.

- Com uso da ontologia a classificação é trabalhada como uma hierarquia de classes, ou seja, a ontologia utiliza a relação **"is-a"**, entre as classes e entre os indivíduos e suas classes. Assim a classe *"Museu"* **"is-a"** classe *"EdificioDeEducacaoECultura"* e esta última **"is-a"** classe de *"Edificacao"*, além disso pela relação entre o indivíduo e classe, temos que a instância *"museuPaulista"* **"is-a"** indivíduo pertencente à classe *"Museu"* o mesmo acontece com as instâncias restantes.

  Logo com a utilização de mecanismos de inferência que consideram o significado da relação **"is-a"**, inferem que as instâncias da classe *Museu* também sejam instâncias da classe pai *Edificacao*, por conseguinte quando o usuário busca "Edificacao"a classe *Edificacao* terá todos os indivíduos de suas subclasses, ver figura 4.10.

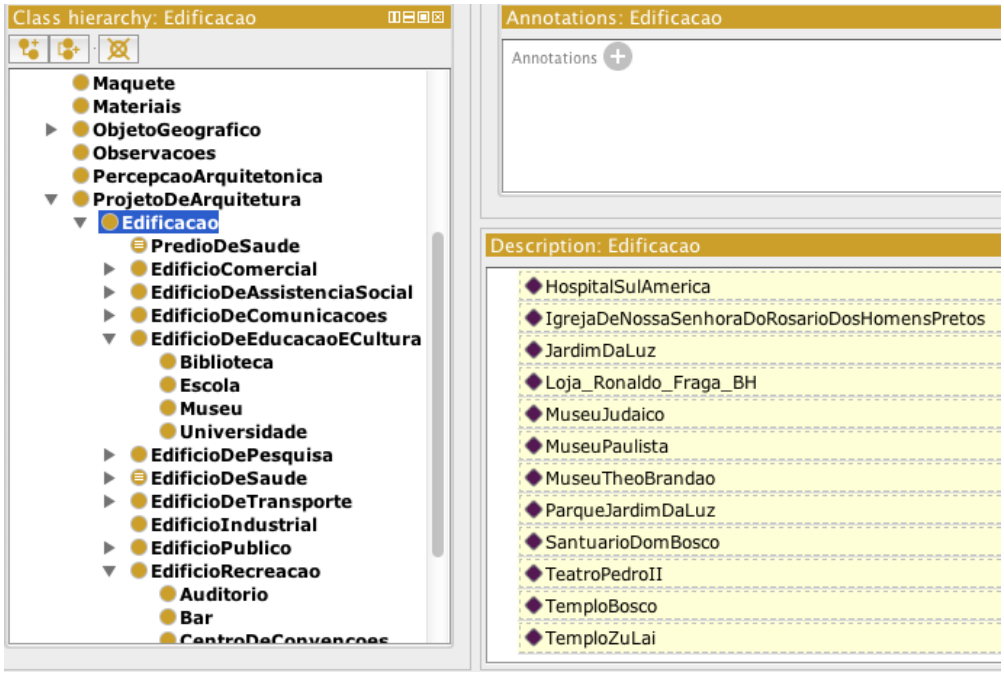


Figura 4.10: *Lista de indivíduos da Classe "Edificacao"como resultado da inferência.*

E através de consultas em SPARQL, pode-se também observar que a classe *Edificacao* possui as instâncias de suas subclasses, ver figura 4.11.

b) Como segundo exemplo na ontologia temos duas instâncias, a primeira *"SantuarioDomBosco"* que pertence à classe *Santuario* e a segunda instância *"TemploBosco"* que pertence à classe *Templo*, ambas instâncias fazem referência à mesma edificação, mas foram registradas com nomes diferentes e cada instância tem relacionadas imagens diferentes, por exemplo a instância

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p:  <http://www.ime.usp.br/~msolis/ontologies/ontoArquigrafia1#>
select distinct  ?classe ?subclasse ?individuos
WHERE{
            ?subclasse rdfs:subClassOf* ?classe.
            ?individuos rdf:type ?subclasse.
            FILTER(?classe=p:Edificacao ).
}
order by ?individuos DESC(?individuos)
```

classe	subclasse	individuos
Edificacao	Universidade	FaculdadeDeHistoriaEGeografiaDaUniversidadeDeSaoPaulo
Edificacao	Hospital	HospitalIsraelitaAlbertEinstein
Edificacao	Hospital	HospitalSulAmerica
Edificacao	Igreja	IgrejaDeNossaSenhoraDoRosarioDosHomensPretos
Edificacao	EdificioIndustrial	JardimDaLuz
Edificacao	Loja	Loja_Ronaldo_Fraga_BH
Edificacao	Museu	MuseuJudaico
Edificacao	Museu	MuseuPaulista
Edificacao	Museu	MuseuTheoBrandao
Edificacao	Parque	ParqueJardimDaLuz
Edificacao	Santuario	SantuarioDomBosco

Figura 4.11: *Resultado da consulta em SPARQL para listar as instâncias da classe "Edificacao"como resultado da inferência.*

"SantuarioDomBosco" tem a propriedade "TemImagem"*SantuarioDomBosco1983View* que é a imagem, e a instância *"TemploBosco"* com a propriedade "TemImagem"que tem as imagens*ImageTemploBosco1 e ImageTemploBosco2.*

Logo pela utilização da linguagem OWL 2 pode-se fazer que as instâncias *"SantuarioDomBosco"* e *"TemploBosco"* sejam declaradas como iguais ao utilizar a sintaxe *owl:SameAs.*
Ao aplicar o mecanismo de inferência presente no Protégé, podemos observar na figura 4.12 que as instâncias das imagens relacionadas a *"SantuarioDomBosco"* incluindo as imagens dele também pertencem à instância *"TemploBosco"* e vice-versa.

Assim quando o usuário consultar quais são as imagens de *"SantuarioDomBosco"*, a ontologia retornará todas as imagens relacionadas à instância *"SantuarioDomBosco"* incluindo todas as imagens relacionadas com a instância *"TemploBosco"*, posto que ambas instâncias *"SantuarioDomBosco"* e *"TemploBosco"* são iguais.
Este resultado pode ser observado na figura 4.13.

Para o caso de um banco de dados tradicional, se tivermos registrado o indivíduo *"SantuarioDomBosco"* relacionado como a imagem *SantuarioDomBosco1983View* e registramos também o indivíduo *"TemploBosco"* como suas imagens a *ImageTemploBosco1 e ImageTemploBosco2.*

Quando o usuário procure quais são as imagens pertencentes ao *"SantuarioDomBosco"* o banco de dados tradicional só retornará todas as imagens pertencentes a ele, que para este exemplo seria a imagem *SantuarioDomBosco1983View.*

c) Como terceiro exemplo, se o usuário busca a frase "Edificações em São Paulo".

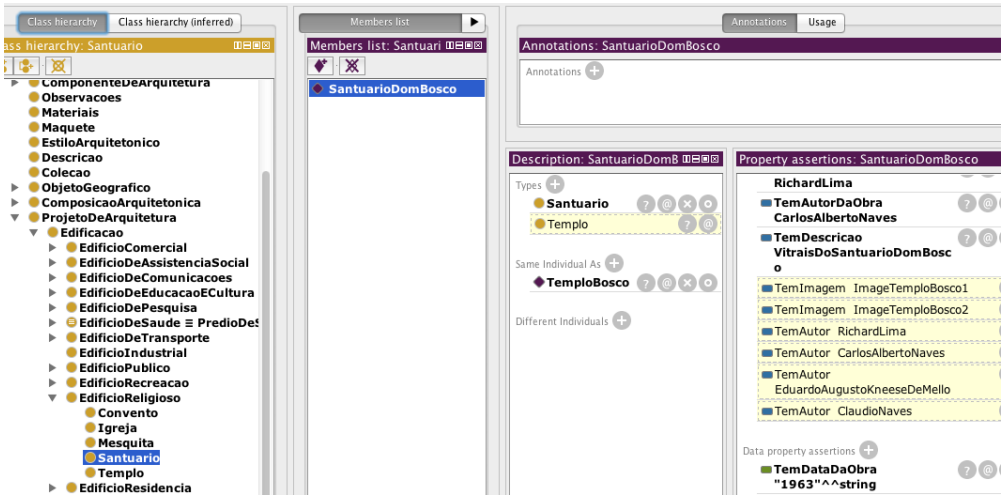


Figura 4.12: *Vemos as imagens relacionadas a "SantuarioDomBosco"incluindo as imagens do "TemploBosco"*

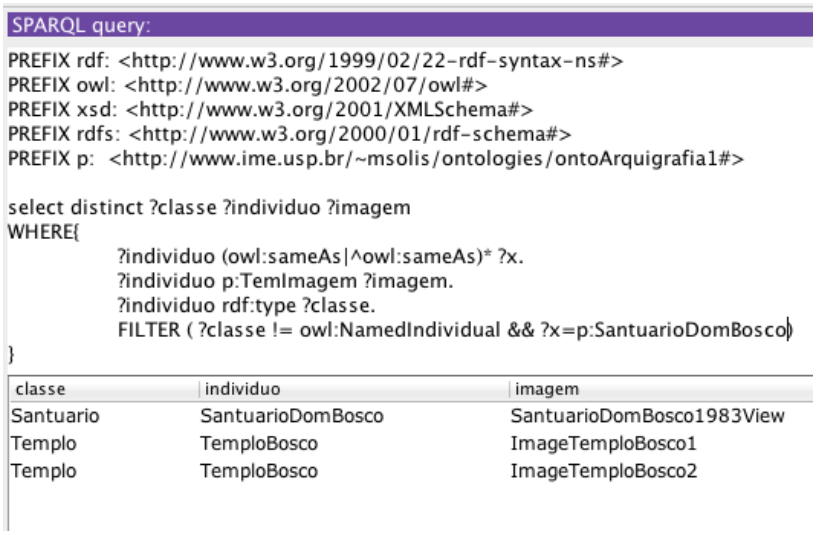


Figura 4.13: *Resultado da consulta da consulta "Quais são as imagens de SantuarioDomBosco?"*

- Se o usuário busca a frase mencionada acima no sistema Arquigrafia, ele terá como resultado muitas imagens, algumas delas estão relacionadas somente a uma parte da frase procurada, ou seja, cada imagem pode ter diferentes etiquetas como *"edificações"*, *"são"*, *"paulo"*, *"são paulo"*, *"edificações em são paulo"*. Desse jeito o resultado dessa busca retornará algumas imagens que podem não ser relevantes para o usuário.

  Na figura(4.14) vemos o resultado sem uso da ontologia na Arquigrafia.

- E com uso da ontologia, os resultados para essa consulta estão mais relacionados à frase completa, devido a que na ontologia existe a propriedade de objeto o relacionamento *"LocalizadoEm"* que filtra aos indivíduos da classe *"Edificacao"* em relação a sua localização.
  Na figura(4.15) vemos o resultado com uso da ontologia na Arquigrafia.

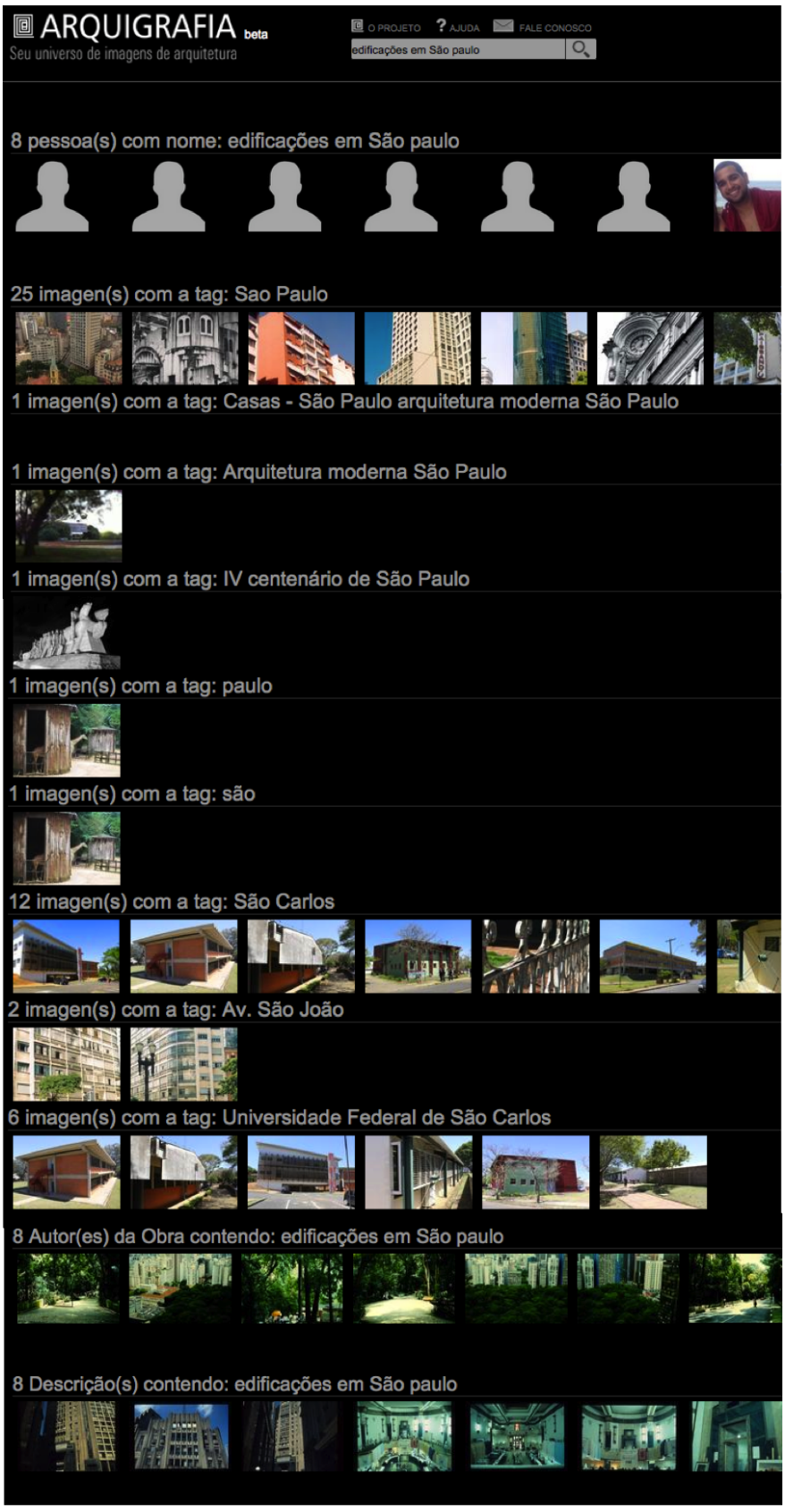


Figura 4.14: *Resultado da consulta "Edificações em São Paulo"em Arquigrafia.*

Portanto, para estes casos específicos podemos dizer que a utilização de ontologias melhora a recuperação de resultados.

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p:  <http://www.ime.usp.br/~msolis/ontologies/ontoArquigrafia1#>
Select ?classe ?nomeEdificacoes ?cidade
WHERE{
?nomeEdificacoes p:LocalizadoEm ?cidade.
?nomeEdificacoes rdf:type ?classe.
FILTER (?classe != owl:NamedIndividual).
FILTER regex(str(?cidade),"Saopaulo","i").
}
```

classe	nomeEdificacoes	cidade
Museu	MuseuJudaico	CidadeSaoPaulo
Teatro	TeatroPedroII	CidadeSaoPaulo
EstacoesMetroviarias	EstacaoJabaquaraDoMetro	CidadeSaoPaulo
Hospital	HospitalIsraelitaAlbertEinstein	CidadeSaoPaulo
Auditorio	AuditorioTribunalDeContas	CidadeSaoPaulo
Igreja	IgrejaDeNossaSenhoraDoRosarioDosHomensPre	CidadeSaoPaulo
Universidade	FaculdadeDeHistoriaEGeografiaDaUniversidadeD	CidadeSaoPaulo
Museu	MuseuPaulista	CidadeSaoPaulo
Biblioteca	BibliotecaBrasilianaGuitaEJoseMindlin	CidadeSaoPaulo
Parque	ParqueJardimDaLuz	CidadeSaoPaulo

Figura 4.15: *Resultado da consulta "Edificações em São Paulo"na ontologia utilizando o SPARQL.*

Capítulo 5

Proposta de Dissertação

Neste capítulo apresentamos a proposta da dissertação, com as atividades planejadas para a sua conclusão juntamente com o cronograma. Para a conclusão do trabalho pretende-se:

a) Melhorar a fase de construção da ontologia, com o apoio de um especialista na área da arquitetura. Em particular, as seguintes atividades estão planejadas:

- Melhorar a taxonomia criando novas classes ou classes sinônimas para apoiar na fase de expansão da consulta.
- Criação de novos relacionamentos e propriedades de objeto ou de tipo.
- Criação de mais indivíduos ou instâncias na ontologia para validar a mesma.

b) Melhorar a fase de indexação e pre-processamento:

- Automatizar a extração de termos da ontologia, e trabalhar com o banco de dados do sistema Arquigrafia :
  - Pretende-se utilizar a API da ferramenta Jena[1] para executar consultas SPARQL e poder recuperar os nomes de todas as classes, os indivíduos para criar a lista de termos que será utilizada pelo índice invertido.
  - Em relação ao banco de dados de Arquigrafia pretende-se acessar a sua informação para recuperar a informação relevante como o título, as etiquetas das imagens, a localização das edificações, a descrição da imagem, os autores das imagens, para guardar essa informação numa base de dados local e poder executar consultas necessárias.
- Criar uma validação de termos presentes no documento da imagem que contém a informação relevante do sistema Arquigrafia como título da imagem, etiqueta da imagem, descrição da imagem e o nome do autor; esta validação analisará cada termo junto com os termos vizinhos para procurá-los na ontologia e trata-los como termos compostos. Além disso, se nos documentos da imagem se encontrar palavras compostas como por exemplo "Espelho-d′agua", não serão divididos e serão considerados como um único termo.

c) Será implementado o algoritmo de expansão de consulta descrito na seção 3.4 para a obtenção de uma nova consulta com termos adicionados que foram extraídos da ontologia para ser utilizada na recuperação das imagens relacionadas à consulta.

d) Para a fase de recuperação de imagem descrita na seção 3.5, será implementado o algoritmo de similaridade entre os documentos das imagens e a consulta expandida.

e) Criar uma interface web com as imagens e informação pertencente a Arquigrafia para avaliar os resultados da busca.

[1] Jena: https://jena.apache.org/

- Pretende-se desenvolver uma aplicação web com a linguagem Java, que utilize a ontologia proposta como base para a expansão de termos na consulta original, para avaliar os resultados da busca considerando os falsos positivos e negativos.
- Também pretende-se fazer consultas ao banco de dados local (com informação relevante do sistema Arquigrafia) sem utilizar a ontologia para poder comparar seus resultados com os resultados acima.

f) Elaboração de um artigo apresentando a metodologia e os resultados.

g) Redação da dissertação.

O cronograma de atividades é apresentado a seguir:

Atividade	**Meses**					
	Oct	Nov	Dez	Jan	Fev	Mar
Iteração para melhora da ontologia	x	x				
Desenvolver uma ferramenta para a extração automática de termos da ontologia e trabalhar com o banco de dados do sistema Arquigrafia para recuperar informação relevante.		x				
Desenvolver uma ferramenta para a validação de termos compostos presentes no documento da imagem.		x	x			
Implementação do algoritmo de expansão de consulta.			x	x		
Implementação do algoritmo para a recuperação das imagens.				x	x	
Desenvolver uma aplicação web que utilize a ontologia proposta para recuperar as imagens relacionadas à consulta.					x	x
Criação de consultas tradicionais ao banco de dados local para avaliar seus resultados e compara-los com a aplicação web.					x	x
Elaboração de artigo para uma conferência.					x	x
Redação da dissertação.		x	x	x	x	x
Defesa.						x

Referências Bibliográficas

Almeida e Bax (2003) Mauricio B Almeida e Marcello P Bax. Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da Informação, Brasília*, 32(3):7–20. Citado na pág. 3

Baeza-Yates e Ribeiro-Neto (1999) R. Baeza-Yates e B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. ISBN 020139829X. Citado na pág. 12

Borst (1997) Willem Nico Borst. *Construction Of Engineering Ontologies For Knowledge Sharing And Reuse*. Tese de Doutorado, Universiteit Twente. Citado na pág. 3

Christiaens (2006) Stijn Christiaens. Metadata mechanisms: From ontology to folksonomy... and back. Em *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, páginas 199–207. Springer. Citado na pág. 10

Daconta *et al.* **(2003)** Michael C. Daconta, Leo J. Obrst e Kevin T. Smith. *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*. Wiley Publishing. ISBN 0471432571. Citado na pág. 3

Díaz-Galiano *et al.* **(2009)** Manuel Carlos Díaz-Galiano, Maite Teresa Martín-Valdivia e LA Ureña-López. Query expansion with a medical ontology to improve a multimodal information retrieval system. *Computers in Biology and Medicine*, 39(4):396–403. Citado na pág. 15

Fernández *et al.* **(2011)** Miriam Fernández, Iván Cantador, Vanesa López, David Vallet, Pablo Castells e Enrico Motta. Semantically enhanced information retrieval: an ontology-based approach. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(4):434–452. Citado na pág. 15

Gruber (1995) Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5):907–928. Citado na pág. 3

Grüninger e Fox (1995) Michael Grüninger e Mark S Fox. Methodology for the design and evaluation of ontologies. Em *International Joint Conference on Artificial Inteligence (IJCAI95), Workshop on Basic Ontological Issues in Knowledge Sharing*. Citado na pág. 8

Guarino (1998) Nicola Guarino. Formal ontology and information systems. Em *Proceedings of the first international conference (FOIS'98)*, páginas 3–15. IOS Press. Citado na pág. 3

Guizzardi (2000) Giancarlo Guizzardi. Desenvolvimento para e com reuso: Um estudo de caso no domínio de vídeo sob demanda. Dissertação de Mestrado, Universidade Federal do Espírito Santo, Brasil. Citado na pág. 3, 8, 16

Hois *et al.* **(2009)** Joana Hois, Mehul Bhatt e Oliver Kutz. Modular ontologies for architectural design. Em *FOMI*, páginas 66–77. Citado na pág. 23

Jackson e Moulinier (2007) Peter Jackson e Isabelle Moulinier. *Natural Language Processing for Online Applications: Text Retrieval, Extraction and Categorization.* John Benjamins Publishing Company. ISBN 9789027249920. Citado na pág. 12

Lin (1998) Dekang Lin. An information-theoretic definition of similarity. Em *ICML-International Conference on Machine Learning*, volume 98, páginas 296–304. Citado na pág. 11, 26

Liu *et al.* **(2006)** Yong Liu, Congfu Xu, Qiong Zhang e Yunhe Pan. Ontology based semantic modeling for chinese ancient architectures. Em *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, volume 21, página 1808. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. Citado na pág. 23

Mai (2008) Jens-Erik Mai. Actors, domains, and constraints in the design and construction of controlled vocabularies. *Knowledge organization*, 35(1):16–21. Citado na pág. 9

Manning *et al.* **(2008)** Christopher D. Manning, Prabhakar Raghavan e Hinrich Schütze. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, USA. ISBN 0521865719, 9780521865715. Citado na pág. 12

McBride (2004) Brian McBride. The resource description framework (rdf) and its vocabulary description language rdfs. Em *Handbook on ontologies*, páginas 51–65. Springer. Citado na pág. 4

Morais e Ambrósio (2007) Edison Andrade Martins Morais e Ana Paula L Ambrósio. Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens. Relatório técnico, Technical report, Universidade Federal de Goiás. Citado na pág. vii, 9

N.I.S. e Organization (2005) N.I.S. e National Information Standards Organization. *Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies*. National information standards series. NISO Press. ISBN 9781880124659. Citado na pág. 9

Salton *et al.* **(1975)** Gerard Salton, Anita Wong e Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620. Citado na pág. 13

Schickel-Zuber e Faltings (2007) Vincent Schickel-Zuber e Boi Faltings. Oss: A semantic similarity function based on hierarchical ontologies. Em *IJCAI*, volume 7, páginas 551–556. Citado na pág. 11

Trillo (2005) Christian Danniel Paz Trillo. Recuperação de vídeos indexados por conceitos. Dissertação de Mestrado, Instituto de Matemática e Estatística da Universidade de São Paulo, 21/03/2005. Citado na pág. 15, 26, 27