

UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
FACULDADE DE FILOSOFIA E CIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA INFORMAÇÃO

Eder Antonio Pansani Junior

**ONTOLOGIAS NO PROCESSO DE INDEXAÇÃO AUTOMÁTICA DE
DOCUMENTOS TEXTUAIS**

MARÍLIA
2016

UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
FACULDADE DE FILOSOFIA E CIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA INFORMAÇÃO

EDER ANTONIO PANSANI JUNIOR

**ONTOLOGIAS NO PROCESSO DE INDEXAÇÃO AUTOMÁTICA DE
DOCUMENTOS TEXTUAIS**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Informação da Faculdade de Filosofia e Ciências - Universidade Estadual Paulista “Júlio de Mesquita Filho” – UNESP, campus de Marília, como requisito parcial para obtenção do título de Mestre em Ciência da Informação.

Área de concentração: Informação, Tecnologia e Conhecimento.

Linha de pesquisa: Informação e Tecnologia.

Orientador: Prof. Dr. Edberto Ferneda.

MARÍLIA

2016

Pansani Junior, Eder Antonio.
P196o Ontologias no processo de indexação automática de documentos
textuais / Eder Antonio Pansani Junior. – Marília, 2016.
126 f. : il. ; 30 cm.

Orientador: Edberto Ferneda.
Dissertação (Mestrado em Ciência da Informação) - Universidade
Estadual Paulista “Júlio de Mesquita Filho”, Faculdade de Filosofia e
Ciências, 2016.
Bibliografia: f. 120-126.

1. Indexação automática. 2. Cabeçalhos de assunto. 3. Ontologias
(Recuperação da informação). I. Título.

CDD 029.5

EDER ANTONIO PANSANI JUNIOR

**ONTOLOGIAS NO PROCESSO DE INDEXAÇÃO AUTOMÁTICA DE
DOCUMENTOS TEXTUAIS**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Informação da Faculdade de Filosofia e Ciências - Universidade Estadual Paulista “Júlio de Mesquita Filho” – UNESP, campus de Marília, como requisito parcial para obtenção do título de Mestre em Ciência da Informação.

Área de concentração: Informação, Tecnologia e Conhecimento.

Linha de pesquisa: Informação e Tecnologia.

BANCA EXAMINADORA

Orientador: Prof. Dr. Edberto Ferneda

Universidade Estadual Paulista “Júlio de Mesquita Filho”
Faculdade de Filosofia e Ciências, Campus de Marília

Prof.a Dra. Mariângela Spotti Lopes Fujita

Universidade Estadual Paulista “Júlio de Mesquita Filho”
Faculdade de Filosofia e Ciências, Campus de Marília

Prof. Dr. Elvis Fusco

Departamento de Ciência da Computação
Centro Universitário Eurípedes de Marília (UNIVEM)

Marília, 6 de maio de 2016.

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me dado saúde, força, sabedoria e paciência para superar as dificuldades e obstáculos que enfrentei.

Agradeço a minha querida esposa Amanda, pela paciência necessária a esta etapa de nossa vida. Agradeço também por me incentivar a buscar meus sonhos, nunca me deixar desistir e compreender e aceitar meus momentos de ausência. Muito obrigado por tudo, você é muito importante em minha vida e eu te amo, muito.

À minha família que sempre me apoiou e ajudou nos momentos em que precisei durante estes anos de estudos e viagens, Pai, Mãe, Vô Antenor, Bê... muito obrigado pela ajuda.

À minha querida sogra e sogro que sempre tentaram me ajudar de todas as formas possíveis, sua preocupação e cuidado foram muito importantes.

Agradeço aos amigos Ricardo e Ubiratan pelo incentivo inicial, pela motivação e por nossas conversas e viagens que foram especialmente importantes.

Ao Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, pelo apoio e incentivo à qualificação que me foi concedido, sem isso talvez eu não teria conseguido.

A todos os amigos que estiveram ao meu lado durante essa caminhada, em especial aos professores do IFSP e os alunos que compreenderam e incentivaram que eu prosseguisse.

Aos amigos que fiz em Marília cursando as disciplinas da pós-graduação, obrigado pela troca de experiências e discussões que foram de grande utilidade.

À Unesp de Marília pela oportunidade que me foi oferecida, onde apesar das dificuldades sempre ofereceu um ambiente propício ao ensino e aprendizagem.

Em especial ao meu Orientador, prof. Dr Edberto, pela paciência e empenho dedicados a elaboração deste trabalho, sempre incentivando e apoiando em todos os momentos, sua participação foi decisiva.

À professora Dra Mariângela Fujita e ao professor Dr. Elvis Fusco por terem aceitado compor a banca e pelo oferecimento de tão ricas colaborações para o aprimoramento deste trabalho.

Enfim, a todos os que, direta ou indiretamente me ajudaram a chegar até aqui...

Obrigado!

Quem sabe, divide.
Quem não sabe, pergunta.

Mario Sergio Cortella

RESUMO

Apesar dos avanços tecnológicos das últimas décadas, a busca por informações relevantes ainda é uma tarefa árdua. A recuperação de informação envolve, por um lado, um acervo documental que deve ser representado por expressões linguísticas que resumem seu conteúdo temático. Por outro lado, pessoas tentam descrever linguisticamente as suas necessidades de informação a fim de obterem documentos relevantes para satisfazer tais necessidades. Um sistema de recuperação de informação é, portanto, um elemento mediador entre um acervo documental e seus requisitantes. Um dos aspectos que interferem diretamente na sua eficiência é a forma como os documentos são representados. Sendo assim, pesquisas sobre indexação automática tomam importância, principalmente em ambiente de grande produção e disseminação de documentos, como é o caso da Web. A utilização de vocabulários controlados como elementos de normalização terminológica é um recurso utilizado para melhorar os resultados do processo de indexação. Este trabalho tem por objetivo propor, avaliar e desenvolver um método de utilização de ontologias no processo de indexação automática de documentos textuais, fazendo uso da estrutura lógica e conceitual das ontologias de domínio e implementado um método que permite aos sistemas de indexação automática a realização de inferências automáticas, favorecendo uma representação dos documentos mais semântica e abrangente. Conclui-se com o estudo que a utilização das ontologias como vocabulários controlados em sistemas de indexação automática pode oferecer resultados promissores, permitindo a descoberta automática de termos e a resolução de alguns problemas ligados à linguagem que permeia todo o processo de recuperação de informação.

Palavras-chave: Indexação Automática; Vocabulário Controlado; Ontologias; Recuperação da Informação;

ABSTRACT

Despite the technological advances of recent decades, the search for relevant information is still an arduous task. The information retrieval involves, on the one hand, a documentary collection that must be represented by linguistic expressions which summarize its thematic content. On the other hand, people try describing linguistically their information needs in order to obtain relevant documents to satisfy those needs. An information retrieval system is therefore a mediating element between a documentary collection and its requesters. One of the aspects that directly interferes in their efficiency is how documents are represented. Therefore, researches on automatic indexing take importance, particularly, in an environment of large production and dissemination of documents, as it's the case of the Web. The use of controlled vocabularies as terminology standardization elements is a feature used to improve the results of the indexing process. This study aims to propose, evaluate and develop a method for using ontologies in the automatic indexing process of textual documents, making use of logical and conceptual structure of domain ontologies and implementing a method that enables automatic indexing systems, an execution of automatic inferences, favoring a semantic and comprehensive documents representation. The study conclusion is that the use of ontologies as controlled vocabularies in automatic indexing systems can offer promising results, allowing the automatic discovery of terms and the resolution of some language related problems that permeates the whole process of information retrieval.

Keywords: Automatic Indexing; Controlled Vocabulary; Ontology; Information Retrieval;

Lista de Figuras

Figura 1 - Representação do processo de recuperação de informação	28
Figura 2 – Exemplo de declaração dos <i>namespaces</i> em uma ontologia OWL	65
Figura 3 – Exemplo de declaração do cabeçalho de uma ontologia OWL.....	67
Figura 4 – Exemplo de definição de uma classe	68
Figura 5 – Exemplo de definição de uma classe usando blocos OWL.....	68
Figura 6 – Exemplo de definição de uma Hierarquia de classes em OWL.....	69
Figura 7 – Exemplo de definição de uma classe equivalente	70
Figura 8 – Exemplo de definição de uma classe disjunta.....	70
Figura 9 – Exemplo de definição de um indivíduo	71
Figura 10 – Exemplo de uma forma alternativa para a definição de um indivíduo	72
Figura 11 – Exemplo de propriedade de objetos	72
Figura 12 – Exemplo de propriedade de dados	73
Figura 13 – Exemplo de definição de uma hierarquia de propriedades	74
Figura 14 – Representação gráfica de classes e propriedades	75
Figura 15 – Exemplo da forma simplificada de utilização da propriedade <i>label</i>	77
Figura 16 – Exemplo da utilização da propriedade <i>label</i> com os parâmetros de idioma.....	78
Figura 17 – Arquitetura da Web Semântica proposta em 2005	80
Figura 18 – Exemplo de consulta SPARQL	81
Figura 19 – Modelo de arquitetura de um sistema de indexação automática.....	84
Figura 20 – Modelo de arquitetura do processo de indexação automática utilizando o método proposto	85

Figura 21 – Página inicial do Repositório de Ontologias BioPortal.....	86
Figura 22 – Cabeçalho da ontologia PedTerm.....	87
Figura 23 – Trecho da ontologia PedTerm, exemplo de criação dos rótulos (<i>labels</i>)	88
Figura 24 – Página inicial do projeto Apache Jena	89
Figura 25 – Execução de uma consulta SPARQL	90
Figura 26 – Página inicial do servidor Apache Jena Fuseki	92
Figura 27 – Página de administração das bases de dados do servidor Apache Jena Fuseki.....	93
Figura 28 – Interface de consultas a base <i>PediatricTerminology</i>	94
Figura 29 – Página inicial do protótipo desenvolvido	97
Figura 30 – Exemplo de Indexação automática a partir de um termo de indexação	101
Figura 31 – Interface do sistema exemplificando o teste explicado na Figura 30.....	102
Figura 32 – Consulta SPARQL para o exemplo de atribuição de conceitos	103
Figura 33 – Exemplo de atribuição de termos sinônimos	105
Figura 34 – Consulta SPARQL para o exemplo de descoberta de sinônimos	105
Figura 35 – Consulta SPARQL para o exemplo de atribuição de conceitos	106
Figura 36 – Interface do sistema exemplificando o teste explicado na Figura 33.....	107
Figura 37 – Exemplo de atribuição de termos sinônimos	107
Figura 38 – Interface do sistema exemplificando o teste explicado na Figura 37.....	108
Figura 39 – Consulta SPARQL para o exemplo descoberta de termos equivalentes	109
Figura 40 – Exemplo de Indexação multilíngue.....	112
Figura 41 – Interface do sistema exemplificando o teste explicado na Figura 40.....	113
Figura 42 – Interface do sistema exemplificando o teste explicado na Figura 40.....	113
Figura 43 – Interface do sistema exemplificando o teste explicado na Figura 40.....	114
Figura 44 – Consulta SPARQL para o exemplo de indexação multilíngue.....	115
Figura 45 – Interface do sistema exemplificando o teste explicado na Figura 40.....	116

Figura 46 – Consulta SPARQL para o exemplo de indexação multilíngue.....	117
--	-----

SUMÁRIO

1	Introdução	13
1.1	Objetivo Geral	18
1.2	Objetivos Específicos	18
1.3	Materiais e Métodos	19
1.4	Trabalhos relacionados	20
1.5	Organização do Trabalho	24
2	Recuperação de Informação	26
2.1	O processo de Recuperação de Informação	28
2.1.1	Documentos (Corpus)	28
2.1.2	Representação dos documentos	29
2.1.3	Usuários	32
2.1.4	Expressão de Busca.....	33
2.1.5	Representação da Expressão de Busca.....	34
2.1.6	Função de Busca	34
2.1.7	Resultado da Busca	34
2.2	Modelos de Recuperação de Informação	35
2.2.1	Modelo Booleano.....	36
2.2.2	Modelo Vetorial.....	37
2.2.3	Modelo Probabilístico	38
3	Indexação Automática	42
3.1	Indexação por extração automática.....	53
3.2	Indexação por atribuição automática	55
3.3	Indexação automática utilizando ontologias.....	56
4	Ontologias	58
4.1	Ontologias e a Ciência da Informação	61
4.2	Web Ontology Language (OWL).....	63
4.2.1	Namespaces.....	65
4.2.2	Cabeçalhos	66

4.2.3	Classes.....	67
4.2.4	Indivíduos	71
4.2.5	Propriedades.....	72
4.2.6	Identificadores e <i>Labels</i>	76
4.2.7	OWL 2.....	78
4.3	SPARQL	79
5	Método de expansão de termos de indexação baseado em ontologia	82
5.1	Conceito	82
5.2	Desenvolvimento	86
6	A utilização do protótipo na expansão de termos	98
6.1	Atribuição de conceitos	100
6.2	Descoberta de termos sinônimos	104
6.3	Indexação Multilíngue	110
7	Considerações finais	118
	REFERÊNCIAS	120

1

Introdução

A necessidade de informações para atender as mais diversas demandas é uma realidade para a maioria dos seres humanos. Buscamos conhecimento para sanar uma curiosidade pontual, para resolver uma situação de trabalho ou para que possamos expandir nossos conhecimentos e assim tornarmo-nos mais capacitados para a vida.

A importância do registro do conhecimento produzido pela humanidade é preocupação desde tempos imemoriáveis. A representação dos conhecimentos em algum suporte original, documentos, que podem ser armazenados, preservados e disponibilizados. Nesse sentido, as bibliotecas são instituições de grande importância, pois, são responsáveis por armazenar, preservar e disseminar estes registros, promovendo assim a multiplicação dos conhecimentos da humanidade.

Com o advento da computação e posteriormente o surgimento da internet, documentos puderam ser armazenados em formato digital e rapidamente disponibilizados para que todos acessem estes registros.

A difusão da tecnologia e do acesso à internet mudou o modo como a sociedade lida com documentos e informações, pois, uma parte significativa destes registros produzidos pela ciência e pela sociedade em geral está em formato digital, e, além disso, grande parte está disponível na Web. Nesse contexto, a Ciência da Informação adquire um importante papel, pois estuda as particularidades do fluxo informacional.

A Ciência da Informação teve origem em meados do século XX, no contexto da revolução científica e tecnológica do período de pós-Segunda Guerra Mundial, marcado pelo crescimento exponencial da produção técnica e científica, a chamada explosão informacional.

Pode-se considerar que os elementos constituintes do processo de recuperação de informação resumem algumas das preocupações da Ciência da Informação.

O processo de recuperação da informação é composto por um acervo de documentos, um usuário com alguma necessidade de informação e um sistema responsável por permitir a seleção de documentos relevantes para este usuário. Estes sistemas devem representar o conteúdo dos documentos do acervo e apresentá-los ao usuário de forma que lhe permita uma rápida seleção dos itens que satisfazem total ou parcialmente suas necessidades informacionais, traduzidas ao sistema por meio de uma expressão de busca.

A eficiência de um sistema de recuperação de informações depende fundamentalmente da similaridade entre a expressão de busca elaborada pelo usuário em comparação à representação dos documentos de um acervo. Esta representação pode ser elaborada pela redação de resumos, catalogação ou ainda pela indexação, sendo a indexação um processo importante e passível de automação o que a torna objeto de muitas pesquisas ao longo das últimas décadas.

Indexar é o ato de selecionar ou definir termos (palavras ou expressões) que irão descrever o conteúdo de um determinado documento, sempre levando em consideração uma clientela (BORGES; MACULAN; LIMA, 2008, p.181).

De acordo com Gil Leiva (2012, p.31):

A indexação gera palavras-chave, índices ou os cabeçalhos de assunto de um documento. Para obtê-los, previamente foi desencadeada uma sucessão interativa e simultânea dos processos mentais que têm a ver com a percepção da informação, da memória e da compreensão. As operações mentais realizadas pelos seres racionais para a recepção seletiva de informação, a sua codificação simbólica e seu armazenamento e recuperação, denomina-se processo cognitivo.

Lancaster (2004) e Gil Leiva (2008) afirmam o processo de indexação é composto por duas etapas essenciais: a **análise e mapeamento**, onde os assuntos do documento são identificados e extraídos para composição do índice e **tradução**, etapa onde os assuntos são submetidos a comparação utilizando um vocabulário controlado, visando padronização e escolha de termos adequados.

Os termos definidos durante a etapa de indexação servirão como pontos de acesso ao conteúdo, mediante os quais um registro será localizado e recuperado no momento da busca. É

importante destacar que a indexação é um processo de suma importância para os sistemas de recuperação de informação, mas, o sucesso da indexação, está ligado à linguagem utilizada.

A linguagem tem um papel fundamental, pois permite que o homem expresse seus pensamentos, ideias e emoções. Pode ser considerada também um meio de preservação e socialização, tornando público todo o conhecimento que foi acumulado por meio também da linguagem. No entanto, em determinadas situações a linguagem pode ser incompleta, ambígua e até limitada, o que pode dificultar a compatibilização entre a linguagem utilizada pelo autor e os usuários do sistema de informação. (NARUKAWA, 2011, p.22-23).

A capacidade em lidar com a linguagem de um documento é um dos fatores que determina o grau de êxito de um sistema, sendo que, outros fatores importantes são o método utilizado e o agente responsável pela execução do processo.

A indexação é um processo que pode ser desempenhado por humanos ou por computadores, sendo que a primeira abordagem conta com a capacidade de compreensão e cognição humana, que é bastante eficiente ao indicar o assunto de um determinado texto após sua leitura.

No entanto, é preciso considerar que a indexação manual sofre de problemas ligados a subjetividade, custo, tempo para realização ou falta de domínio do idioma por parte do indexador (BORGES; MACULAN; LIMA, 2008, p.183), sendo que, há ainda a inconsistência entre diferentes indexadores ou para o mesmo indexador em diferentes momentos (BORKO, 1977, *apud* GUEDES, 1994) resultando em uma redução na qualidade da indexação e consequentemente nos resultados do sistema de recuperação.

Uma solução utilizada para promover uma maior uniformidade no processo é o uso de vocabulários controlados, definidos por Lancaster (2004, p.19) como “uma lista de termos autorizados”. A aplicação de vocabulários controlados influencia nos processos de indexação e recuperação da informação, ampliando o acesso às ideias do autor e servindo como elo entre a informação criada e a necessidade dos indivíduos que potencialmente necessitam desta (NARUKAWA, 2011, p.23).

A representação da informação realizada através da adoção de terminologias padronizadas resultou em avanços na qualidade da indexação, reduzindo a polissemia e as ambiguidades características da linguagem natural. No entanto, o volume de documentos produzido pela sociedade, aliado a relativa facilidade de sua disseminação via internet, podem

colaborar para a disponibilização de documentos sem uma indexação adequada, o que pode provocar desorganização nas informações disponíveis e, conseqüentemente, dificuldades na recuperação da informação relevante.

As propostas de indexação automática surgiram visando solucionar problemas como o volume de conhecimento produzido. Originalmente ligadas à área de computação, estas propostas exploram a possibilidade de utilizar algoritmos e máquinas para realizar a indexação.

O uso do computador elimina a subjetividade inerente ao ser humano e permite que grandes volumes de documentos possam ser indexados em um curto espaço de tempo, sendo ainda, uma solução menos custosa do ponto de vista financeiro. Além de que, uma máquina é capaz de processar um volume maior do que um humano em um mesmo espaço de tempo.

Várias pesquisas surgiram há algumas décadas buscando aprimorar a indexação automática, utilizando métodos estatísticos, frequência de palavras, posição das palavras no documento, uso de palavras-chave e muitas outras. No entanto, uma das dificuldades da indexação automática é que vários métodos utilizam como insumo inicial termos extraídos do próprio documento, tornando-os assim suscetíveis às limitações da linguagem.

Uma possível solução é a utilização de vocabulários controlados, tais como os cabeçalhos de assuntos ou tesauros, ambos extensamente utilizados na indexação manual, mas complexos de se aplicar aos sistemas automáticos devido a sua construção não ser específica para uso em sistemas computacionais.

Narukawa (2011) ressalta isso em sua pesquisa sobre o uso de tesauros no sistema de indexação automática SISA, onde após criteriosa análise a autora aponta alguns problemas ligados à utilização dos tesauros como vocabulários controlados em sistemas automáticos, sendo que algumas das principais dificuldades são as diferenças entre singular e plural, termos grafados de formas distintas nos documentos e no tesauro ou ainda conceitos implícitos

A autora conclui afirmando que “o uso do tesauro no sistema de indexação automática SISA oferece benefícios, mas que as falhas, tornam seu uso ainda pouco confiável”, destacando ainda que os problemas apontados podem ser minimizados pela adoção de tratamentos morfológicos. No entanto, outra alternativa é a substituição do tipo do vocabulário controlado, como as ontologias, que com suas relações pode contornar as dificuldades apontadas e atender as demandas por sistemas capazes de oferecer tratamento semântico de informações.

As ontologias utilizadas na área da computação, são aparatos capazes de representar o conhecimento, as relações entre os conceitos, hierarquia, restrições e vários constructos que oferecem uma semântica elaborada para representar um determinado domínio do conhecimento, além de serem construídas seguindo uma estrutura formal e processável por computadores.

A recuperação de informação baseada em ontologia (*ontology-based information retrieval*) é um campo de pesquisa da área de Ciência de Computação. Um grande número de dissertações e teses abordam uma diversificada gama de propostas e abordagens para a utilização de ontologias no processo de recuperação de informação.

A utilização de ontologias como instrumento de controle terminológico em sistemas computacionais de indexação automática pode permitir que estes sistemas sejam capazes de fazer inferências sobre o documento e os termos dele extraídos utilizando para isso as relações contidas na ontologia de domínio. Outro fator positivo do uso de ontologias é seu suporte a múltiplos idiomas, recurso que permite uma única ontologia servir para indexar documentos de diferentes idiomas.

A disponibilização de documentos na Web indexados automaticamente com uso de ontologias pode melhorar significativamente a capacidade de recuperação de informações. A exemplo da proposta da indexação multilíngue, que proporcionaria a possibilidade de repositórios de informação praticamente universais, pois poderiam ser acessados por termos de vários idiomas diferentes.

As ontologias como ferramentas auxiliares no processo de recuperação de informação vem ao encontro dos objetivos da Web Semântica, que propõem que as informações estejam organizadas de forma que os computadores consigam processá-las e interpretar seus significados. Utilizando as tecnologias propostas pela Web Semântica, tais como *Resource Description Framework* (RDF), *Resource Description Framework Schema* (RDFS), *Ontology Web Language* (OWL) espera-se atender as recomendações do W3C e colaborar para a construção de uma Web mais inteligente e conseqüentemente, mais eficaz e dinâmica.

Neste trabalho, propõem-se um método que utiliza ontologias de domínio como vocabulários controlados para expansão automática de termos de indexação, explorando o fato de que as ontologias são processáveis por computadores é possível desenvolver uma solução capaz de ser incorporada a sistemas de indexação automática.

O processo de indexação automática segue basicamente as mesmas etapas do processo padrão de indexação, agregando etapas posteriores a extração de termos do documento processamentos sintáticos, morfológicos e semânticos (GIL LEIVA, 2008). Na sequência é feita a etapa de tradução, onde cada candidato a descritor é submetido à comparação utilizando um vocabulário controlado, é neste ponto que o método proposto por esta pesquisa pode ser agregado ao processo de indexação, pois ele apresenta uma abordagem que permite ao sistema de indexação automática efetuar buscas no vocabulário controlado de forma mais inteligente, tais como a descoberta de termos relacionados, equivalentes, sinônimos e em outros idiomas que a ontologia suporte.

A solução proposta ataca a problemática apontada por Narukawa (2011), que faz críticas ao processo de indexação automática devido a falhas na etapa de cotejamento dos termos candidatos junto ao vocabulário controlado, devido principalmente a este não ser adequado ao processamento automático, desta forma, o emprego de ontologias da solução proposta pode ser uma solução adequada para esta dificuldade.

Utilizando a linguagem de consulta SPARQL e algumas ontologias de domínio já construídas e disponibilizadas na internet, são executados alguns testes no protótipo desenvolvido, permitindo assim demonstrar que, as propostas de atribuição de novos termos, indexação multilíngue e resolução de sinônimos ou termos equivalentes são factíveis no contexto apresentado, proporcionando ainda resultados preliminares que podem produzir novas pesquisas e evoluções na área de indexação de informação automática.

1.1 Objetivo Geral

Esta pesquisa **tem por objetivo** propor e desenvolver um método de utilização de ontologias em uma das etapas do processo de indexação automática de documentos textuais.

1.2 Objetivos Específicos

- Compreender o processo de recuperação de informações e os elementos que o compõe.
- Analisar o processo de indexação automática, seus tipos e fatores intervenientes.

- Compreender o conceito e a utilização das ontologias como ferramentas de controle terminológico.
- Implementar um protótipo de expansão automática de termos baseado em ontologias, como parte do processo de indexação automática.
- Analisar os resultados e emitir um parecer sobre a proposta e suas possíveis contribuições para as pesquisas em indexação automática.

1.3 Materiais e Métodos

Esta pesquisa propõe um método de expansão de termos de indexação baseado em ontologias de domínio, sendo que este pode ser agregado ao processo de indexação, junto a etapa de tradução dos conceitos obtidos a partir do processo de análise de assunto. Desta forma é correto afirmar que o método proposto é uma ferramenta que visa auxiliar o processo de indexação automática de documentos textuais.

Primeiramente foi providenciada uma ontologia de domínio, que é utilizada como vocabulário controlado neste trabalho, desta forma é preciso tornar este corpo de conhecimento acessível ao sistema. Para isso, a ontologia OWL foi inserida dentro do servidor Apache Jena Fuseki. Assim as consultas enviadas pelo protótipo via linguagem PHP podem utilizá-la como vocabulário controlado.

A etapa seguinte foi desenvolver um protótipo de sistema que é capaz de aceitar um termo como entrada, o idioma deste e qual o tipo de relação a ser buscada no vocabulário, tal como busca de termos equivalentes ou hierarquicamente genéricos. É importante frisar que os termos de entrada não são extraídos de nenhum documento, o protótipo simula a etapa de indexação seguinte a extração automática dos candidatos a descritores.

A partir de termos previamente extraídos dos documentos textuais durante o processo de indexação pretende-se por meio do uso da linguagem SPARQL criar consultas que sejam capazes de fazer inferências em ontologias OWL, promovendo a descoberta automática de novos termos.

Para o desenvolvimento do trabalho foi implementado uma solução computacional que utiliza como materiais algumas tecnologias e ferramentas listadas em ordem de utilização no desenvolvimento:

- Apache Jena Framework: é um *framework* Java para se construir aplicações de Web Semântica. Ele fornece um ambiente de programação para RDF, RDFS, OWL, SPARQL.
- Apache Jena Fuseki: é um servidor SPARQL que provê um mecanismo de execução de consultas em linguagem SPARQL sobre o protocolo HTTP.
- LAMP: é uma combinação de softwares livres e de código aberto. LAMP é um acrônimo que se refere as primeiras letras de Linux Apache MySQL e PHP, sendo este um componente bastante comum na área de desenvolvimento e programação de aplicações para Web.
- sparqllib.php: é uma biblioteca que permite consultas em SPARQL a partir da linguagem de programação PHP.

Detalhes mais técnicos sobre a construção do protótipo, seu funcionamento e a apresentação dos resultados são abordados nos Capítulos 5 e 6.

A presente pesquisa pode ser caracterizada como exploratória, pois, tem como objetivo principal propor e desenvolver um método de expansão de termos de indexação utilizando ontologias de domínio como vocabulário controlado. Para isso foi desenvolvido um protótipo computacional capaz de permitir a execução de testes que visam provar que o método é factível.

1.4 Trabalhos relacionados

Algumas pesquisas da área de ciência da informação que abordam o tema de recuperação da informação utilizando ontologias são apresentadas neste tópico, visando compreender e apresentar as propostas atuais de sistemas de indexação automática. Trabalhos como o CIRI - *Concept-based Information Retrieval Interface* (AIRIO *et al*, 2004) não tratam especificamente de sistemas para indexação automática, no entanto, abordam a proposta do uso de ontologias em alguma etapa do processo de recuperação de informações, sendo por esta razão pertinentes a fundamentação apresentada nesta pesquisa.

O sistema CIRI apresentado por (AIRIO *et al*, 2004, tradução nossa) utiliza um modelo de dados de três camadas, que tem por objetivo representar as relações entre os conceitos de uma forma mais natural à recuperação de informação. Os níveis de abstração apresentados pelo modelo são: nível conceitual, no qual os conceitos são estruturados hierarquicamente utilizando

as ontologias; nível linguístico, que contém expressões que correspondem aos conceitos; e o nível de ocorrência, onde estão algumas instâncias que se relacionam com o segundo nível, sendo que, estas ocorrências, podem ser por exemplo, palavras simples, compostas ou obtidas por lematização¹ ou stemização².

Este sistema propõe melhorias na interface de recuperação de textos baseada na construção automática de consultas com o auxílio de ontologias. Seu funcionamento baseia-se em um algoritmo que utiliza de um conjunto inicial de termos e as relações da ontologia para realizar a expansão.

Outro sistema que utiliza ontologias é o Zstation, descrito por Bräscher (2002) onde o enfoque do estudo é a resolução dos problemas ocasionados pelo fenômeno da ambiguidade. O Zstation é um sistema de tratamento automático da linguagem natural, que faz a análise automática da sentença mediante a coleta de informações, relacionadas tanto à sentença quanto às propriedades semânticas e morfológicas das palavras, grupos de palavras, frases e conexões possíveis entre eles, até que o conhecimento coletado permita propor uma ou várias interpretações.

Para realizar o processamento, o sistema inclui ferramentas linguísticas, como um dicionário automático, uma gramática morfológica, uma gramática de argumentos e uma ontologia, que especifica a relação temática definida num argumento (BRÄSCHER, 2002). Conclui-se assim que o Zstation utiliza ontologias para fazer o tratamento sintático-semântico das informações, e, outros recursos como o processamento de linguagem natural e aplicação de dicionários de lemas que o tornam uma pesquisa relevante para a área de recuperação de informação.

Outra proposta de indexação automática é apresentada por Amorim e Cheriaf (2007), no qual um sistema de recuperação de informações baseado em ontologias é desenvolvido buscando melhorar a precisão e revocação em um sistema da área de arquitetura, engenharia e construção. O sistema apresentado pelos autores tem como enfoque melhorar a recuperação de informação relevante por meio do aprimoramento tanto na indexação quanto na interface de busca.

¹ Original *lemmatisation*

² Original *stemming*

Este sistema utiliza a base de documentos do INFOHAB³ - Centro de Referência e Informação em Habitação, para criar um repositório de índices destes documentos. Seu funcionamento consiste em, inicialmente extrair dos documentos os termos mais representativos (descritores), efetuar a remoção de palavras irrelevantes (*stopwords*), a normalização morfológica (*stemming*), e por fim, a ponderação de cada descritor com base na frequência inversa dos documentos.

Utilizando uma ontologia proveniente de um sistema denominado ONTOARQ, desenvolvido por Amorim e Rabelo (2006), são descobertas as relações associativas dos termos descritores, gerando assim, um banco de índices que permite consultas também submetidas aos tratamentos linguísticos e a expansão com suporte do ONTOARQ, pois, segundo os autores, “o pequeno número de argumentos de busca digitados pelo usuário talvez seja um dos maiores problemas para obter-se uma maior eficácia do sistema”.

Amorim e Cheriaf (2007) apresentam resultados obtidos por meio da comparação da quantidade de documentos recuperados ao informar algumas palavras chave no mecanismo de busca original do INFOHAB e no sistema por eles desenvolvido. Os resultados indicam que o sistema desenvolvido pelos autores proporciona uma melhora na eficácia da recuperação da informação, quantitativamente em 35% de precisão média.

Outro trabalho, apresentado por Narukawa (2011), tem por objetivo identificar as características que definem e distinguem os tipos de vocabulários controlados, analisar as propostas metodológicas e os sistemas de indexação automática. Por meio da aplicação do vocabulário controlado Thesaurus Agrícola Nacional (ThesAgro) no *Sistema de Indización SemiAutomatica* (SISA) a autora faz uma análise comparativa entre a indexação manual da Biblioteca Nacional de Agricultura (BINAGRI) e os resultados deste experimento, analisando os fatores intervenientes e os problemas relacionados à indexação automática.

A conclusão obtida pela autora é que o vocabulário controlado aplicado no sistema SISA foi elaborado para a indexação manual, e, portanto, considera que seu uso pode ocasionar problemas ligados ao tratamento morfossintático em situações de diferenças de flexão de número e gênero, uso de símbolos e outras. A autora afirma ainda que a substituição dos tesauros por ontologias é uma iniciativa importante a ser investigada.

³ <http://www.infohab.org.br/>

Kara et al. (2012) apresenta um sistema de recuperação de informações baseado em ontologias usando uma abordagem de indexação semântica, o sistema proposto implementa as etapas de extração de informação baseada em palavras chave utilizando ontologias de domínio para realizar inferências e melhorar a recuperação de informações. Métodos de expansão de consulta são propostos e colocadas à prova em comparação com métodos tradicionais, sendo que o estudo conclui que a utilização de indexação semântica pode resolver problemas simples de ambiguidades estruturais.

Outra abordagem que estuda a recuperação semântica baseada em ontologias é proposta por Lijun e Xu (2011), que apontam uma solução para os mecanismos tradicionais de busca baseados apenas na correspondência de palavras chave sem suporte a semântica. Os autores afirmam que a utilização destes sistemas depende muito do domínio da aplicação pelos usuários, sendo essa uma grande limitação.

Sistemas de recuperação semântica baseados em ontologias de domínio podem efetuar comparações semânticas, garantindo buscas mais abrangentes e precisas. A utilização das ontologias na etapa de processamento da expressão informada pelo usuário pode auxiliar na eliminação das ambiguidades da linguagem natural e também melhorar a precisão por meio da descoberta das reais necessidades dos usuários. Outros benefícios apontados são a facilidade da expansão das buscas e a descoberta de conteúdo relacionado ao assunto (LIJUN; XU, 2011).

Os autores finalizam o estudo afirmando que o sistema desenvolvido permite aos usuários efetuar buscas utilizando a linguagem natural e obter resultados satisfatórios, concluindo assim que a utilização de ontologias no processo de recuperação de informações pode melhorar a semântica e produzir resultados mais precisos.

Em uma abordagem mais preocupada com a etapa de indexação, Castells, Fernandez e Vallet (2007) discorrem em seu trabalho sobre uma adaptação do modelo espaço vetorial para recuperação de informações baseada em ontologias.

Os autores explicam que, uma das grandes motivações da web semântica é a busca semântica. Muitos trabalhos dedicam-se a estudar e propor formas de agregar inteligência as buscas, melhorando seus mecanismos em grandes repositórios de documentos.

A proposta dos autores consiste em um mecanismo de anotações semiautomáticas nos documentos e um modelo de recuperação de informações, que utiliza o modelo espaço vetorial para criar um algoritmo capaz de lidar com diferentes valores de relevância para os termos,

resultando em um sistema com buscas semânticas que apresenta os resultados ordenados por relevância. Os resultados dos experimentos realizados em um corpus relativamente grande mostram que a abordagem apresenta uma clara melhora nas buscas por palavras chave (CASTELLS; FERNANDEZ; VALLET, 2007).

1.5 Organização do Trabalho

Este primeiro capítulo tem por objetivo apresentar e contextualizar o tema deste trabalho, assim como explicitar os objetivos do mesmo.

No segundo capítulo, explora-se a recuperação de informação, abordando as etapas principais do processo e os modelos de recuperação de informação mais utilizados, concluindo com o apontamento do modelo mais adequado à uma possível solução que utilizaria a proposta deste trabalho.

O terceiro capítulo fala sobre a indexação automática. Inicialmente, abordam-se os conceitos da indexação, os processos e os instrumentos. O foco deste capítulo é descrever a indexação automática como solução possível ao crescente aumento no volume de documentos. São descritos ainda, os dois tipos de indexação automática: por extração e por atribuição de termos, para que seja possível compreender como utilizar ontologias como ferramentas de apoio aos processos de indexação apresentados.

No quarto capítulo, discorre-se sobre o termo ontologia, conceituando seu significado para as ciências da Informação e da Computação. Posteriormente, conceitua-se a linguagem para construção de ontologias OWL, explicando suas particularidades e elementos básicos. Cabe frisar que são descritos apenas alguns aspectos da OWL devido principalmente à sua complexidade, buscando-se sempre ilustrar situações e aplicações com exemplos, para garantir o entendimento.

No quinto capítulo descreve-se com mais detalhes o funcionamento do método e os materiais utilizados no desenvolvimento do protótipo. A aplicação criada não leva um nome, pois não se trata de um sistema de indexação automática, mas sim de uma demonstração de uma etapa que pode vir a ser aplicada um sistema de indexação automática.

O sexto capítulo é dedicado a apresentar os exemplos e propostas do uso de ontologias na indexação automática. Iniciamos explicando superficialmente sobre como foram feitos os

testes, tratando sobre cada uma das situações e apresentando a teoria e os benefícios esperados do uso da ontologia, a execução dos testes e os resultados obtidos.

No sétimo e último capítulo, são discutidos os avanços provenientes do uso de ontologias em comparação à proposta de uso de tesauros como vocabulário controlado para indexação automática. São apresentadas também as conclusões sobre a pesquisa e propostas para possíveis trabalhos futuros.

Recuperação de Informação

O termo “recuperação de informação” faz referência a um processo de busca de informações que satisfaçam as necessidades informacionais de um indivíduo. A ação de recuperar informações pode ser comumente conhecida como uma “pesquisa”, que informalmente pode ser compreendida com a ação de um indivíduo que utiliza um sistema ou ferramenta que faz buscas sobre um acervo qualquer objetivando assim encontrar informações (textuais, imagéticas ou sob qualquer outro formato) que satisfaçam suas necessidades de informação.

Para Manning, Raghavan e Schütze (2009) o significado do termo recuperação de informação pode ser muito extenso, como por exemplo “pegar o cartão de crédito na carteira e verificar seu número pode ser uma forma de recuperação de informação”. No entanto, os autores colocam que em um ambiente acadêmico o termo recuperação de informação pode ser definido assim:

Recuperação de Informação (RI) é encontrar material (geralmente documentos) de natureza não estruturada (geralmente textos) que satisfaçam as necessidades de informação a partir de grandes coleções (geralmente armazenadas em computadores) Manning, Raghavan e Schütze (2009, p. 38)

Para Ferneda (2012, p.13) o processo de recuperação de informação “consiste em identificar no conjunto de documentos (corpus) de um sistema, quais atendem à necessidade de informação do usuário”, corroborando assim com a definição de Manning, Raghavan e Schütze.

As definições de RI podem sofrer pequenas variações dependendo da área de pesquisa, na computação os bancos de dados com seus respectivos sistemas gerenciadores são responsáveis por permitir consultas por meio de uma linguagem específica. Nestes ambientes a recuperação de informação pode ser compreendida como o processo de apresentar todos os

objetos que satisfaçam a uma consulta que contém uma expressão de busca. Na ciência da informação o princípio básico é satisfazer as necessidades informacionais do usuário, sendo assim recuperar informações significa recuperar um conjunto de documentos que contém informações relevantes à necessidade do indivíduo que interage com o sistema.

Antes de prosseguir com argumentação acredita-se que é importante contextualizar os termos “informação” e “documento” utilizados neste capítulo e no decorrer do trabalho.

A palavra informação pode ser utilizada em diversos contextos, além de poder assumir diferentes significados e definições, dependendo de quem a usa, sendo que dentro da ciência da informação o termo pode ser definido como:

[...] um conhecimento inscrito (gravado) sob a forma escrita (impressa ou numérica), oral ou audiovisual. A Informação comporta um elemento de sentido. É um significado transmitido a um ser consciente por meio de uma mensagem [...]. Essa inscrição é feita graças a um sistema de signos (a linguagem) [...] Le Coadic (2004).

A partir das proposições de Le Coadic (2004) pode-se afirmar que uma informação é algo palpável, inscrito em algum suporte e que transmite uma mensagem ao indivíduo que é capaz de compreender a linguagem ao qual a informação está inscrita.

Uma segunda definição, de Shannon e Weaver (1949, p.3) diz que “[uma informação] é o que acrescenta algo a uma representação [...]”. Recebemos informação quando o que conhecemos se modifica. ”. Nesta definição percebe-se um ponto de vista distinto, onde os autores entendem por informação algo menos palpável. Aqui a informação é entendida como um processo cognitivo, algo intangível, diferente da proposta de Le Coadic, onde a informação seria uma “coisa”.

Em uma abordagem bastante prática Buckland (1991) traz uma proposta atual do termo informação. Em seu texto *Information as Thing* o autor identifica três usos do termo:

- **Como processo** – quando alguém é informado, o que ele sabe é alterado, ou seja, o ato de informar ou a comunicação do conhecimento ou notícias sobre um fato ou ocorrência;
- **Como conhecimento** – o que é percebido pela informação enquanto processo, o conhecimento comunicado. Sua principal característica é a intangibilidade;
- **Como coisa** – aquilo que é visto como informativo: objetos, documentos, textos, dados ou eventos. A sua principal característica é a tangibilidade, sua materialidade.

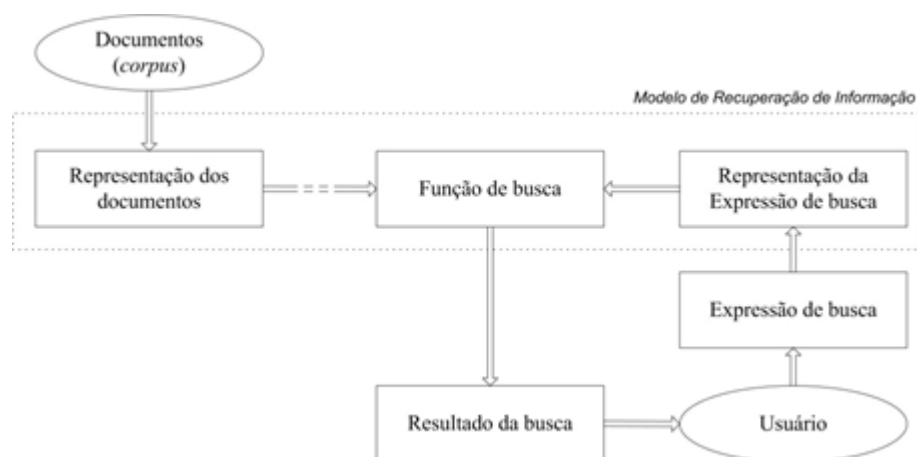
Segundo essa definição os dois primeiros usos são relacionados a processos, necessitando de algum suporte material para o conhecimento inscrito, já no terceiro uso Buckland define a “informação como coisa” tal como Le Coadic, sendo esta definição a mais adequada aos propósitos desta pesquisa.

Adota-se então que informação é algo palpável, um suporte físico para os conhecimentos, tal como um documento, desta forma “informação” e “documento” são utilizados como sinônimos no decorrer deste estudo.

2.1 O processo de Recuperação de Informação

Ferneda (2012, p.14) propõe uma representação simplificada do processo de recuperação da informação, que é apresentada na Figura 1. Esta representação norteia os estudos no campo de RI, criando subáreas que se complementam. Nas seções seguintes cada elemento do processo é abordado de forma breve.

Figura 1 - Representação do processo de recuperação de informação



Fonte: Adaptado de Ferneda, 2012, p.14

2.1.1 Documentos (Corpus)

Documento é a parte material, os registros onde os dados estão armazenados, o objeto de interesse de qualquer busca. Na literatura da ciência da informação pode-se encontrar algumas diferentes definições do termo documento, no entanto, adota-se para este trabalho o conceito de “informação como coisa” definido por Buckland (1991), onde documento seria algum objeto que contém informação.

Esta abordagem está em consonância com a definição de Le Coadic (2004, p.5), que coloca que um documento é:

O termo genérico que designa os objetos portadores de informação. Um documento é todo artefato que representa ou expressa um objeto, uma ideia ou uma informação por meio de signos gráficos e icônicos (palavras, imagens, diagramas, mapas, figuras, símbolos), sonoros e visuais (gravados em suporte de papel ou eletrônico).

É importante destacar ainda a problemática dos documentos disponíveis em ambientes digitais, hoje reproduzir os conhecimentos produzidos pela sociedade é algo bastante simples e rápido, pois a internet permite que documentos sejam armazenados nos computadores e compartilhados em âmbito global, o que na prática é uma evolução importante é também uma preocupação de importância ímpar para a CI, visto que a complexidade em representar adequadamente estes documentos é primordial para que seja possível recuperar tais conhecimentos quando se fizer necessário. Ferneda (2012, p.15) destaca que “no contexto da Web esta desvinculação dos documentos da forma física tradicional (como o papel) possibilita uma integração entre diferentes suportes, como textos, imagens, sons e outros recursos” que na prática formam documentos bastante interessantes do ponto de vista informacional, porém ao mesmo tempo, desafiadores do ponto de vista da recuperação de informação.

2.1.2 Representação dos documentos

A representação das informações contidas nos documentos é uma etapa importante no contexto do processo de recuperação de informação, pois é nesta etapa que são catalogadas as informações da obra e criado o conjunto de termos que servirá para indexar o seu conteúdo.

De acordo com Maimone; Silveira e Tálamo (2011) a representação da informação (RI) pode ser subdividida em representação descritiva e temática. A primeira descreve as características específicas do documento, denominada descrição bibliográfica. Ela permite a sua individualização e também define e padroniza os pontos de acesso responsáveis pela busca e recuperação do documento. Já a segunda se restringe a representação dos assuntos do documento, se referindo aqui ao conteúdo do mesmo.

Esta seção, assim como a proposta deste estudo se dedicam a estudar a representação temática, por esta razão a representação descritiva não será abordada.

A etapa de representação da informação busca descrever os documentos de forma que seja possível recuperá-los posteriormente. Esta tarefa é feita descrevendo o conteúdo do

documento utilizando procedimentos de indexação e seu objetivo é criar formas de facilitar a recuperação do documento com base em uma expressão de busca, sendo que os documentos recuperados sejam o mais úteis quanto possível para o usuário que os procura.

Novellino (1996) afirma que a indexação é um dos processos típicos da representação temática da informação, seguida do processo de classificação e da elaboração de resumos. A autora coloca então que o processo de RI pode ser resumido em dois passos principais:

1. Análise de assunto de um documento e a colocação do resultado desta análise nome exposição linguística;
2. Atribuição de conceitos ao documento analisado.

A realização desta fase pressupõe uma linguagem documentária, instrumento de padronização da indexação, a qual visa garantir que os indexadores de um mesmo sistema usem os mesmos conceitos para representar documentos semelhantes. Ele também é um instrumento de comunicação ao permitir que os indexadores e os usuários partilhem de um mesmo vocabulário.

Após a análise do processo de representação das informações contidas em um documento pode-se compreender sua importância frente ao processo de recuperação da informação, pois os resultados apresentados em um sistema ou mecanismo de busca são consequência da etapa de indexação. Se um documento for indexado de forma medíocre as chances de sua recuperação são reduzidas, independentemente da relevância de seu conteúdo.

Segundo Fujita (2012, p.13) a indexação e a recuperação da informação possuem uma estreita relação, como uma relação de causa e efeito. Isto se dá, pois, a indexação possui características que provocam efeitos sobre a recuperação. A especificidade é uma delas, pois a escolha de termos específicos, que serão traduzidos em descritores igualmente específicos para que posteriormente o usuário possa fazer busca por palavras chave específicas. Esta é apenas uma das decisões que devem ser tomadas durante a indexação e que refletem diretamente no resultado da recuperação.

Discorrendo sobre a etapa de indexação, Albrechtsen (1993) propõe algumas concepções metodológicas sobre a representação da informação, sendo uma delas orientada ao conteúdo, onde a análise dos assuntos dos conteúdos dos documentos envolve a identificação de tópicos ou assuntos que não são explicitamente colocados na estrutura textual superficial do documento, mas que são prontamente perceptíveis a um indexador. Esta estratégia possui um potencial de maximizar os assuntos tratados do documento, podendo resultar em uma melhoria na recuperação do mesmo.

No entanto é necessário refletir sobre a logística do processo de indexação, pois ter indexadores humanos responsáveis por esta tarefa pode torna-la inexecutável visto o volume crescente de conhecimento produzido pela sociedade.

Uma possível solução para este problema é o uso de métodos automáticos de indexação, onde computadores utilizam algoritmos e processos pré-programados para indexar os documentos, viabilizando assim indexar um grande volume de documentos. Os métodos de indexação automáticos são objetos de pesquisa há anos e são explicados com mais detalhes no 42.

As discussões apresentadas até o momento permitem concluir que o processo de indexação, seja realizado de forma manual ou automática é bastante subjetivo e envolve uma série de decisões. Além disso ao considerar a indexação realizada por humanos é possível que diferentes indivíduos executem uma mesma tarefa de forma distinta, ocasionando uma despadronização no processo.

Desta forma para nortear os processos decisórios e promover uma maior uniformidade nos processos de indexação é necessário que se estabeleçam normas, que no contexto da biblioteconomia e da CI são chamadas de Políticas de Indexação.

Uma política de indexação pode ser definida de forma resumida como “um parâmetro para tomada de decisão e tem a função de orientar o processo de indexação de acordo com as necessidades de informação dos usuários de um determinado sistema, considerando todo o contexto envolvido” (PICKLER, 2014, p.38).

Segundo Carneiro (1985, p.165):

Na implantação de um serviço de indexação há que se levar em conta um grande número de variáveis que irão afetar consideravelmente o desempenho do serviço. Definir essas variáveis, estabelecer princípios e critérios que servirão de guia na tomada de decisões para otimização do serviço, racionalização dos processos e consistência das operações nele envolvidas, são objetivos de uma política de indexação.

Em uma definição mais atualizada e abrangente Fujita (2012, p.17) afirma que:

A política de indexação não deve ser vista como uma lista de procedimentos a serem seguidos, e sim um conjunto de decisões que esclareçam os interesses e objetivos de um sistema de informação e, particularmente, do sistema de recuperação de informação. A política decide não só sobre a consistência dos procedimentos de indexação em relação aos efeitos que se necessita obter na recuperação, mas, principalmente, sobre a delimitação de cobertura temática em níveis qualitativos e quantitativos tendo em vista os domínios de assuntos e as demandas dos usuários.

A partir da definição de Fujita (2012, p.17) acredita-se que a política da indexação discorre “sobre a delimitação de cobertura temática em níveis qualitativos e quantitativos” logo é possível concluir que a definição de políticas tem uma importância ímpar no que diz respeito a indexação, pois é seguindo os critérios e princípios definidos na política que a indexação pode ter resultados mais uniformes e precisos em relação ao contexto e os usuários do sistema de informação.

A indexação que é o grande tema deste estudo deve então ser pautada por políticas de indexação, pois estas definem critérios como a exaustividade, a especificidade, a correção e a coerência. Todos estes fatores são abordados com mais profundidade no 42.

2.1.3 Usuários

O próximo elemento a ser discutido é o usuário, parte fundamental do processo de RI, pois é sua necessidade informacional que motiva as pesquisas na área de RI, proporcionar sistemas melhores e mais eficientes, com capacidade para satisfazer estas necessidades é um objetivo norteador das pesquisas em RI.

Segundo Riecken (2006, p. 55):

O usuário e os conteúdos estão no centro das preocupações da CI e não as tecnologias. Entretanto, a CI pode valer-se de práticas de modelagem e notação oriundas da engenharia de software e da computação visando facilitar a operacionalização em sistemas computacionais, suas bases de dados e redes, a partir dos modelos e representações conceituais.

O ponto de vista apresentado por Riecken mostra a importância do usuário para a ciência da informação além de reforçar uma abordagem também defendida por Ferreira (1995), que define as novas abordagens como “centradas no usuário”, isto na prática significa que as interfaces de busca, as políticas de indexação e o uso em geral dos sistemas de recuperação de informação vem sendo pensados considerando o usuário e suas capacidades cognitivas e sociais como parte singular do processo. Isto em sempre foi assim, pois em alguns modelos de indexação apenas dados estatísticos ou cabeçalhos de assuntos eram usados para indexar os conjuntos de documentos, não se importando com a experiência dos utilizadores.

Dentro desta abordagem centrada no usuário a recuperação da informação é um processo de produção de sentido por parte do usuário, o qual utiliza a informação para construir conhecimento (FERNEDA, 2012, p.18), sendo que para Morris (1994) a informação é

parcialmente construída pelo usuário durante este processo de produção de sentido, e só existe fora dele de maneira incompleta. Portanto, os sistemas de RI devem ser modelados considerando as necessidades informacionais e o comportamento do usuário.

2.1.4 Expressão de Busca

O próximo elemento do processo de RI é a expressão de busca, ligada diretamente ao usuário a capacidade de formular uma expressão de busca adequada define o sucesso ou o fracasso em recuperar informações relevantes. Ferneda (2012, p. 18) define uma expressão de busca como:

[...] o meio que o usuário utiliza para comunicar ao sistema sua necessidade informacional. É composta geralmente por termos que podem ser especificados em linguagem natural ou por meio de algum tipo de linguagem artificial dependendo dos recursos do sistema. Para se obter sucesso neste processo os usuários necessitam de ter um mínimo de conhecimento do tema de interesse e do seu vocabulário do domínio.

Nesta etapa temos um problema delicado, quando o autor diz que “para se obter sucesso neste processo os usuários necessitam de ter um mínimo de conhecimento do tema de interesse e do seu vocabulário do domínio” cria-se uma grande dificuldade, pois se um usuário busca alguma informação sobre algo, possivelmente ele não tem muito conhecimento sobre este assunto em específico, no entanto, se ele não tem conhecimento sobre este assunto ele não terá condições de identificar quais termos podem ter sido usados para representar os documentos que irão satisfazer suas necessidades informacionais, criando uma espécie de loop complexo de se resolver.

Considerando esse problema temos que a etapa de criação da expressão de busca é primordial no resultado do processo de RI, sendo que a importância de pesquisas e novas abordagens nesta etapa é muito relevante.

Apontamos aqui uma proposta de uso de ontologias como ferramenta de auxílio a construção da expressão de busca, como os propósitos desta pesquisa limitam-se a etapa de representação dos documentos essa proposta será pouco explorada. Esta proposta será melhor explorada na seção sobre o uso das ontologias para expansão e melhora da expressão de busca.

2.1.5 Representação da Expressão de Busca

A representação da expressão de busca diz respeito a como a expressão construída pelo usuário é comparada com os termos usados para indexação dos documentos. Ferneda (2012, p.19) destaca que muitos sistemas de recuperação de informações oferecem uma série de recursos que facilitam a elaboração da expressão de busca, como, por exemplo, permitir que os usuários insiram frases completas com palavras de pouca relevância, num segundo momento o sistema gera automaticamente uma representação desta expressão apresentando somente os termos que sejam úteis a recuperação da informação, maximizando assim os resultados da busca.

O núcleo desta etapa é garantir que:

[...] independentemente dos recursos oferecidos pelo sistema é necessário que a expressão de busca seja representada de forma similar a utilizada na representação dos documentos. Essa homogeneidade permitirá a comparação entre a busca e todos os documentos do corpus do sistema por meio da função de busca (FERNEDA, 2012, p.19)

2.1.6 Função de Busca

O elemento central do processo de recuperação de informação é a função de busca, que de acordo com Ferneda (2012, p.19) é responsável por “comparar as representações dos documentos com a representação da expressão de busca” retornando assim todos os itens que atendem a estes critérios que “supostamente fornecerão a informação que o usuário procura”.

Em linhas gerais a função de busca calcula de alguma forma o grau de similaridade entre a expressão de busca e cada um dos documentos do corpus, este processo pode dependendo do método adotado servir para atribuir um peso para os documentos recuperados permitindo uma ordenação por relevância, ou ainda apenas dividir o corpus em documentos que são similares e serão apresentados no resultado e documentos que não são similares, sem uma ordenação percebida. Este segundo método é notadamente menos desejável, porém figura em alguns sistemas de RI.

2.1.7 Resultado da Busca

O produto final do processo de RI, “o resultado da busca é um conjunto de documentos que supostamente serão úteis para o usuário, em geral é apresentando como uma lista ordenada pelo grau de importância ou relevância dos materiais em relação a expressão de busca do

usuário” (Ferneda, 2012, p.19). Espera-se como resultado da busca uma massa de documentos que permita ao usuário verificar se estes lhe são úteis, pois o fato de um documento figurar na lista de resultados não necessariamente significa que ele é útil ao usuário ou mesmo relacionado ao contexto da expressão de busca formulada.

Além da imprecisão na recuperação há outros fatores que podem tornar um documento integrante do resultado irrelevante para o usuário, tal como ser obsoleto demais ou mesmo já ter sido previamente recuperado pelo usuário, tornando-o inútil do ponto de vista de alguém que busca informações inéditas.

2.2 Modelos de Recuperação de Informação

Na Figura 1 há três blocos rodeados por uma linha tracejada, o conjunto formado por estes três elementos é chamado por Ferneda (2012) de Modelo de Recuperação de Informação.

De forma resumida pode-se afirmar que a “Representação dos documentos” é o bloco que representa o resultado da indexação do *corpus*, a “Representação da Expressão de busca” traduz as necessidades informacionais dos usuários do sistema e a “Função de busca” é o bloco responsável por comparar e apresentar a intersecção entre as necessidades informacionais dos usuários e os documentos disponíveis no *corpus* que hipoteticamente possuem as informações necessárias para atender estas necessidades. Conclui-se que a eficiência de um sistema de recuperação de informações está ligada ao modelo que ele utiliza.

Alguns dos modelos de recuperação de informação foram criados há algumas décadas, entretanto, suas principais ideias ainda estão presentes na maioria dos sistemas de recuperação atuais e também nos mecanismos de busca da Web (FERNEDA, 2012, p.20).

O fato de estes modelos serem utilizados até hoje nos mostra como seus conceitos ainda são úteis e relevantes, sendo os modelos Booleano, Vetorial e Probabilístico considerados “modelos clássicos de recuperação da informação” (SILVA; SANTOS; FERNEDA, 2013).

Logo, são abordados respectivamente nas subseções 2.2.1, 2.2.2 e 2.2.3 os modelos Booleano, Vetorial e Probabilístico, conceituando cada um deles e abordando suas principais vantagens e desvantagens concluindo com a indicação do modelo mais adequado para os propósitos deste estudo.

2.2.1 Modelo Booleano

O Modelo Booleano é baseado na álgebra booleana, um sistema originado das pesquisas de George Boole, que definiu um sistema de símbolos e regras aplicáveis a números binários e também a enunciados (Ferneda, 2012, p. 21). O sistema de números binários é a base dos sistemas digitais, é graças a ele que os equipamentos digitais como computadores, smartphones e até a própria internet existe.

A álgebra booleana baseia-se em um sistema que pode adotar dois valores: verdadeiro ou falso (1 ou 0), sendo que no contexto de um Sistema de Recuperação de Informações temos que o corpus de documentos é filtrado por uma função de busca em dois conjuntos, o conjunto dos documentos que atendem a expressão de busca, que teoricamente são relevantes e os demais, que são considerados irrelevantes.

O modelo funciona da seguinte forma, os termos usados para se indexar os documentos do corpus são comparados aos termos presentes na expressão formulada pelo usuário, que pode conter operadores lógicos como OR, AND ou NOT. A título de exemplo, considere que um usuário informa apenas um termo e realiza uma busca, todos os documentos que foram indexados por este termo serão recuperados e apresentados como resultado. Uma segunda situação o usuário acontece quando o usuário deseja recuperar todos os documentos que contenham o termo A e o termo B, ele então deverá informar a expressão “A AND B” para que o sistema possa retornar apenas os documentos que foram indexados tanto pelo termo A quanto pelo termo B.

Kuramoto (2002) afirma que o modelo booleano é um dos mais utilizados na recuperação da informação devido a sua simplicidade e formalismo claro, porém os resultados obtidos por este modelo são dotados de pouca precisão. A crítica de Kuramoto ao modelo baseia-se na inabilidade do modelo em fornecer diferentes importâncias aos diferentes termos da busca e consequentemente aos resultados, implicando assim em uma impossibilidade de ordenação dos documentos resultantes na busca. Ferneda (2012) corrobora com todas estas limitações e destaca ainda que tal limitação torna-se mais evidente nos mecanismos de busca da Web, onde uma quantidade muito grande de documentos está disponível para busca, o que aumenta a importância de uma forma de ordenação dos resultados.

O autor ainda levanta outras limitações deste modelo, tais como:

- Dificuldade de formular a expressão de busca, criar uma expressão de busca complexa, capaz de usar operadores booleanos mais específicos pode ser uma tarefa bastante complexa para um usuário que não tem sólidos conhecimentos em álgebra booleana.
- Pouco controle sobre a quantidade de documentos recuperados, é impossível prever quantos registros corresponderão a uma busca, sendo necessárias reformulações a expressão de busca para atingir uma quantidade de documentos aceitável.
- O resultado da busca é composto por todos os documentos que são potencialmente úteis a necessidade do usuário, de acordo com o modelo todos os documentos recuperados possuem igual importância, sendo impossível criar uma ordenação por utilidade ou relevância.

Esta breve explanação sobre o modelo booleano permite concluir que sua utilidade é grande, sua implementação pode ser simples e seus resultados precisos, no entanto, sua incapacidade de lidar com relevâncias diferentes para os termos, tanto no processo de indexação como na comparação com a expressão de busca o tornam uma opção pouco viável para os propósitos deste estudo. Neste trabalho propõem-se usar ontologias para encontrar novos termos, sendo estes por sua vez menos significativos que os termos originais. Implementar estas ideias computacionalmente só é possível utilizando alguma forma de ponderação entre os termos e seus respectivos pesos.

2.2.2 Modelo Vetorial

O Modelo Vetorial também conhecido como Modelo Espaço Vetorial propõe uma abordagem onde é possível obter documentos que respondem parcialmente a uma expressão de busca. Isto é possível associando valores de relevância diferentes aos termos tanto da indexação quanto da expressão de busca formulada pelo usuário. Com o uso deste recurso é possível calcular a similaridade entre a expressão de busca e cada documento do corpus, permitindo que seja apresentado um resultado ordenado pelo grau de similaridade de cada documento em relação a busca (FERNEDA, 2012, p.31).

Quanto ao funcionamento Kuramoto (2002, p.2) explica que:

Enquanto o Modelo Booleano baseia-se na comparação exata entre os termos de uma consulta e aqueles presentes nos documentos, o modelo Vetorial baseia-se na comparação parcial entre a representação dos documentos e a da consulta do usuário.

Isto é possível graças à atribuição de pesos aos termos de indexação presentes na consulta e aqueles presentes nos documentos. Essa ponderação, atribuída a cada termo de indexação, permite calcular o grau de similaridade entre um documento e uma consulta. Portanto, a cada documento que atende a uma consulta é atribuído um grau de similaridade, possibilitando, ao sistema de recuperação de informação, apresentar os resultados de uma consulta de forma ordenada, normalmente, em ordem decrescente dos respectivos graus de similaridade. Trata-se de uma maneira de evidenciar a relevância de cada documento em relação a uma dada consulta. Esses pesos são calculados a partir da frequência de ocorrência do referido termo no documento.

Na prática um documento é representado por um vetor onde cada elemento representa o peso, ou relevância do respectivo termo de indexação para o documento. Os vetores descrevem a posição deste material em um espaço multidimensional, onde cada termo de indexação representa um eixo, cada peso é normalizado com valores entre um (1) e zero (0), sendo os valores mais próximos de um (1) indicam termos com mais importância para a descrição do documento. A expressão de busca funciona da mesma forma, sendo representada por um vetor numérico com os pesos dos termos que a representam, sendo possível então o cálculo de similaridade entre os vetores da expressão e de cada documento.

Dado o funcionamento deste modelo pode-se perceber que ele é adequado em situações com um grande corpus envolvido, devido à sua capacidade de ordenação dos resultados torna-se uma opção útil a mecanismos de busca na Web, além de ser uma opção adequada aos propósitos deste trabalho, pois, é possível que os termos provindos da ontologia sejam agregados aos documentos e a expressão de busca, inferindo pesos a estes sem necessidade de ajustes ou alterações na dinâmica do modelo.

2.2.3 Modelo Probabilístico

Este modelo tem suas raízes na teoria matemática das probabilidades, considerando o espaço amostral como o corpus de documentos e a recuperação de documentos relevantes como um evento possível. Souza (2006) explica que esse modelo supõe que exista um conjunto ideal de documentos que atende a cada uma das possíveis buscas que podem ser feitas no sistema. Uma característica deste modelo é considerar o processo de recuperação de informação algo incerto quanto a relevância, sendo mais realista ao pensar em uma probabilidade de relevância em vez de uma relevância exata, como nos modelos booleano e vetorial (Ferneda, 2013, p.76).

O processo de RI deste modelo é semelhante ao modelo vetorial, pois é baseado em uma expressão de busca inicial, composta por termos aos quais o usuário tenta expressar suas

necessidades informacionais, após submetida ao sistema, esta expressão serve para que seja feito um cálculo da similaridade destes termos com cada documento do corpus, estes valores de similaridade servem ainda para apresentar os resultados de forma ordenada de acordo com uma provável relevância. Após esta etapa do processo o usuário pode selecionar alguns documentos recuperados dentro do resultado inicial, submetendo novamente uma busca agora com estes documentos selecionados para que o sistema possa fornecer uma busca mais precisa, pois ele tem agora alguns documentos que são verdadeiramente úteis ao usuário (Ferneda, 2013, p.77). O autor destaca que este processo é denominado *relevance feedback*, e que ele pode ser repetido até que o usuário se sinta satisfeito com os resultados.

Souza (2006) afirma que este modelo pode utilizar na tentativa inicial técnicas de outros modelos, como o vetorial, e que depois das sucessivas interações e do feedback do usuário a busca torna-se cada vez mais próxima do conjunto ideal, sendo este o maior destaque deste modelo, considerar o usuário como fator decisivo para alcançar os resultados ideais.

Considerando o funcionamento do modelo probabilístico, onde é possível ter uma ordenação dos resultados das buscas, são utilizados termos para indexação do corpus e também na elaboração da expressão de busca, temos que este modelo também é um candidato em potencial para uso neste trabalho, pois a proposta de agregar termos a indexação por meio de uma ontologia cabe dentro deste método.

A ideia de usar uma ontologia como uma ferramenta externa que pode agregar termos e maximizar a capacidade de representação dos documentos baseia-se nas conclusões de Ferneda (2013, p. 77) quando ele afirma que:

Os três modelos clássicos compartilham um mesmo paradigma de representação no qual os termos extraídos dos documentos e das buscas são suficientes para se efetivar o processo de recuperação de informação, considerando-o um sistema fechado no qual o significado dos elementos lexicais é dado pelas suas inter-relações no interior de um corpus documental.

Após abordar brevemente de cada um dos modelos clássicos de recuperação de informação **conclui-se** que o modelo vetorial seria o mais adequado para ser utilizado no desenvolvimento de um sistema de recuperação de informações que utiliza a proposta deste trabalho para expansão dos termos de indexação. Ressalta-se que esta pesquisa não tem por objetivo o desenvolvimento de um sistema de recuperação de informações.

O modelo Booleano não seria um candidato interessante, pois não é capaz de lidar com valores de relevância para cada termo de indexação, sendo assim uma opção inviável para o método proposto.

Quanto ao modelo probabilístico, que também aceita valores de relevância individual para os termos de indexação, não foi a opção indicada devido a seu maior destaque estar ligado a participação do usuário para definição do vetor mais adequado, o que implica em uma situação delicada ao se considerar a proposta de indexação automática. Sem a participação de humanos no processo o modelo probabilístico torna-se uma opção pouco viável.

A principal razão é que no modelo vetorial existe um conjunto de termos que representa cada documento, sendo que estes podem assumir diferentes valores de relevância quanto a representação do documento. Isto é adequado para o método proposto, pois, a expansão dos termos originais por meio da descoberta de termos hierarquicamente genéricos ou sinônimos irá agregar novos descritores ao documento, no entanto, estes não estão explícitos em sua superfície textual, implicando em um valor de relevância menor para estes termos provindos do vocabulário controlado em relação aos termos extraídos diretamente do documento.

O método proposto para uso do modelo vetorial funcionaria da seguinte forma, os termos extraídos do documento teriam peso igual a 1, pois tem relevância máxima na representação do documento. Os termos sinônimos poderiam ter relevância igual a 0,8, um termo mais genérico do que o termo em questão também poderia ter uma relevância igual a 0,8, e um termo ainda mais genérico, poderia ter um peso de 0,6 de relevância e assim sucessivamente.

Esta é apenas uma situação hipotética onde alguns pesos foram escolhidos para ilustrar o funcionamento do método, não sendo valores obtidos a partir de pesquisas ou testes práticos, no entanto permitem compreender como o modelo vetorial pode ser uma opção adequada as propostas deste método.

A indexação é um processo de importância ímpar em relação ao processo de recuperação de informações, pois, uma representação temática coerente reflete diretamente no desempenho do sistema, independente da qualidade e tamanho do acervo ou da função de busca utilizada.

Pode-se definir a indexação como a síntese dos conteúdos dos documentos pertencentes ao acervo, responsável por apresentar os assuntos de forma a permitir que estes sejam utilizados

como pontos de acesso aos quais os utilizadores do sistema podem submeter suas necessidades informacionais traduzidas por meio de uma expressão de busca

Assim, no próximo capítulo são abordados os conceitos da indexação automática, visando explorar as formas de se realizar este processo e também a teoria envolvida na indexação automática utilizando ontologias.

Indexação Automática

A indexação é um procedimento que tem por objetivo a elaboração de índices de assunto dos documentos, sendo parte integrante do processo conhecido como análise documentária.

Entende-se por análise documentária ou análise documental “o conjunto de procedimentos efetuados com o fim de expressar o conteúdo de documentos, sob formas destinadas a facilitar a recuperação da informação” (CUNHA, 1989, p.40).

De acordo com Fujita (1999b, p. 21)

A análise documentária é operacionalmente um tratamento documentário de conteúdo com a finalidade de elaborar representações condensadas do que está contido em textos. Essas representações condensadas são resumos e índices, sendo que esses últimos caracterizam-se mais como pistas de conteúdo. Todo o conjunto de procedimentos para a elaboração de índices de assunto, seja com palavras ou símbolos alfanuméricos, estão inseridas no que se denomina “Indexação”

Kobashi (1994, p.23) afirma que tanto a indexação quanto o resumo são considerados representações condensadas do conteúdo dos documentos, sendo elaborados através de um conjunto de três operações: análise, síntese e representação, constituindo a “análise documentária”. A autora ainda define cada uma dessas três operações:

- Análise: leitura e segmentação do texto para identificação e seleção dos conceitos;
- Síntese: construção do texto documentário com os conceitos selecionados: enunciado de assunto ou resumo;
- Representação: a representação possui duas naturezas distintas, na primeira a representação é construída através de um processo de condensação intensiva do texto original, gerando os diferentes tipos de resumo; na segunda, a

representação é realizada através do uso de uma linguagem documentária, que tem como função a normalização das unidades significantes ou conceituais presentes no texto original: indexação;

Ainda discorrendo sobre a conceituação da indexação e seu posicionamento dentro da análise documentária, Silva e Fujita (2004, p.136) explicam que

para alguns teóricos, principalmente ingleses e norte-americanos, a Indexação é a própria análise documentária, composta das mesmas etapas operacionais com o objetivo de representação do conteúdo informacional de documentos para a elaboração de índices.

Há ainda outro ponto de vista, pois segundo os teóricos espanhóis

a análise documentária comporta dois níveis de divisão: o da forma – análise descritiva ou bibliográfica – refere-se ao tratamento físico da informação, ligado com o suporte. E o do conteúdo, que se refere ao tratamento temático da informação e destina-se à representação condensada do assunto intrínseco ou extrínseco de um determinado documento (SILVA; FUJITA, 2004, p.137).

Desta forma percebe-se uma sutil diferença entre as correntes francesa e espanhola, onde para os primeiros a análise documentária refere-se somente ao tratamento do conteúdo do documento, não adotando uma divisão entre forma e conteúdo, ou descrição física e temática do documento.

Por fim, Silva e Fujita (2004, p.137) afirmam que a corrente inglesa, representada por autores como Foskett, Lancaster, Van Slype, entre outros, faz uso da palavra indexação, entendendo-a como um processo.

Segundo Borko e Bernier (1978, p.8 *apud* Silva e Fujita, 2004, p.137) a indexação é definida como o processo de analisar o conteúdo informacional dos registros do conhecimento e sua expressão na linguagem do sistema de indexação.

Na concepção de Lancaster (2004, p. 8-9) o processo de indexação compreende duas etapas:

- **Análise conceitual do assunto:** requer a compreensão do conteúdo quanto ao tema e tipo de texto para possibilitar a identificação de todos os conceitos que representam o documento visando uma posterior seleção das informações, segundo os parâmetros de exaustividade e especificidade exigidos pelo sistema de indexação. O autor observa que para efetuar uma análise conceitual adequada, o indexador necessita

não apenas compreender o assunto de que trata o documento, mas também as necessidades informacionais dos usuários do sistema.

- **Tradução dos conceitos em termos padronizados:** essa etapa requer a organização das descrições padronizadas de acordo com os termos previamente estabelecidos pelo vocabulário de indexação. Importante salientar que a compatibilidade entre a terminologia de indexação e a de recuperação é fundamental para que os resultados de uma busca sejam satisfatórios.

Após esta breve conceituação sobre a análise documentária e indexação, pode-se afirmar que essa pesquisa considera a indexação como uma forma de representação temática da informação, etapa responsável por criar os índices de assuntos que fazem parte dos sistemas de recuperação de informação.

No entanto, é importante deixar claro que o foco deste capítulo não é estudar a metodologia da indexação, mas sim os instrumentos utilizados neste processo, com destaque as ontologias, que aqui são apresentadas como vocabulários controlados processáveis por computador e úteis em determinadas etapas do processo de indexação automática por atribuição.

Na Figura 1 do capítulo sobre recuperação da informação é apresentado uma representação do processo de recuperação de informação, onde um dos elementos é a “representação dos documentos”, etapa do processo que contém dentre suas tarefas a indexação. Tal afirmação é corroborada por Lancaster (2004, p.6) afirma que:

[...] a indexação de assuntos e a redação de resumos são atividades intimamente relacionadas, pois ambas implicam a preparação de uma representação do conteúdo temático dos documentos. [...] O principal objetivo do resumo é indicar do que se trata o documento ou sintetizar seu conteúdo. Um grupo de termos de indexação serve ao mesmo propósito.

O autor ainda destaca que os termos atribuídos ao índice durante a etapa de indexação servem como pontos de acesso para que o sistema possa localizar no acervo os assuntos de interesse dos usuários, traduzidos por meio de uma expressão de busca. Desta forma deduz-se que o sucesso desta etapa influencia diretamente os resultados de um sistema de recuperação de informações.

Chaumier (1980, *apud* Silva e Fujita, 2004) afirma que “a indexação é a parte mais importante da análise documentária”, pois é ela quem condiciona o valor do sistema. O autor

afirma que “uma indexação inadequada ou insuficiente representa 90% das causas essenciais para a aparição de “ruídos” ou “silêncios”. Entende-se por “ruídos” documentos que fazem parte dos resultados, porém não tem relação com o assunto desejado, e, “silêncios” documentos relevantes à necessidade informacional, mas não recuperados por alguma razão.

De acordo com Novellino (1996):

A principal característica do processo de representação da informação é a substituição de uma entidade linguística longa e complexa - o texto do documento - por sua descrição abreviada. O uso de tal sumarização não é apenas uma consequência de restrições práticas quanto ao volume de material a ser armazenado e recuperado. Essa sumarização é desejável pois sua função é demonstrar a essência do documento. Ela funciona então como um artifício para enfatizar o que é essencial no documento considerando sua recuperação, sendo a solução ideal para organização e uso da informação.

Em sua dissertação, Pickler (2014, p.32) afirma que “indexar consiste em analisar um documento para identificar seu assunto e representar esse assunto em termos descritores que serão traduzidos na linguagem do sistema para padronização”.

A indexação é um processo caracterizado por Fujita (1989, p.120) como:

Uma operação delicada que lida com as ideias a serem transmitidas por palavras que as representem, e depende de variáveis subjetivas submetidas a uma análise de um indexador humano que também possui ideias próprias. Cada palavra de um texto carrega um significado e uma função dentro do contexto e uma mesma palavra empregada em diferentes contextos terá diferentes significados. Quando uma palavra, dita significativa, é extraída isoladamente de um texto estará sendo dissociada de um contexto que lhe impõe um significado e uma função. Uma palavra usada ao acaso, sem obedecer a um rigor contextual, pode transformar a ideia original do autor do documento e produzir uma outra.

A partir da afirmação de Fujita percebe-se que o processo de indexação deve ser feito por profissionais especialistas em tal ofício, pois a atribuição de um termo fora de contexto pode prejudicar a recuperação de um documento ao invés de maximizar seu potencial de recuperação no acervo.

Outro fator que tem forte influência no processo de indexação é a etapa de **tradução**, que segundo Lancaster (2004, p.18) pode ser definida como “[...] a segunda etapa da indexação de assuntos, envolve a conversão da análise conceitual de um documento num determinado conjunto de termos de indexação”.

O autor explica que a etapa de tradução pode ser executada de duas formas distintas, dependendo do tipo da indexação. Na indexação por extração palavras ou expressões do próprio

documento são utilizadas para representar o conteúdo temático, já na indexação por atribuição são utilizados termos provenientes de alguma fonte que não é o próprio documento, sendo frequentemente utilizados os vocabulários controlados (Lancaster, 2004, p.18).

No decorrer deste trabalho adota-se a definição de Lancaster (2004, p. 19) que define um vocabulário controlado como “uma lista de termos autorizados”. O autor explica que o indexador somente pode atribuir a um documento termos que estiverem presentes na lista adotada pela instituição, porém ele afirma que um vocabulário controlado é mais que uma mera lista, pois ele inclui, em geral, uma forma de estrutura semântica, destinada especialmente a:

- Controlar sinônimos, optando por uma forma única padronizada, com remissivas de todas as outras;
- Diferenciar homógrafos. Por exemplo, PERU (país) é um termo bastante diferente de PERU (ave);
- Reunir ou ligar termos cujos significados apresentem uma relação mais estreita entre si.

A proposta do uso de vocabulários controlados permite que os indexadores possam utilizar esta ferramenta para auxiliar na tarefa de indexação, minimizando possíveis falhas inerentes a subjetividade da interpretação humana. Lopes (2002, p.47) elenca algumas vantagens de seu uso:

- Controle total do vocabulário de indexação, minimizando os problemas de comunicação entre indexadores e usuários;
- Com o uso de um tesauro e suas respectivas notas de escopo, os indexadores podem assinalar mais corretamente os conceitos dos documentos;
- Se bem construído, o vocabulário controlado poderá oferecer alta recuperação e relevância e, também ampliar a confiança do usuário diante de um possível resultado negativo;
- As relações hierárquicas e as remissivas de vocabulário controlado auxiliam tanto o indexador, quanto o usuário na identificação de conceitos relacionados;
- Redução no tempo de consulta à base, pois a estratégia de busca será mais bem elaborada com o uso de tesauro.

No entanto, o autor supracitado é justo ao elencar possíveis desvantagens no uso dos vocabulários controlados, tais como:

- Custos: a produção e manutenção da base de dados terão despesas maiores com a equipe de indexadores. Será necessário ainda manter pessoal especializado para atualização do tesouro;
- O vocabulário controlado poderá não refletir adequadamente os objetivos do produtor da base caso esteja desatualizado;
- Um vocabulário controlado poderá se distanciar dos conceitos adequados para a representação das necessidades dos usuários;
- Necessidade e treinamento no uso dos vocabulários controlados tanto para os intermediários, quanto para os usuários finais;
- Desatualização do vocabulário poderá conduzir a falsos resultados.

Em relação ao vocabulário adotado pelo sistema, Vieira (1996, p.13) observa alguns fatores importante a serem considerados:

A atualidade terminológica e proximidade à linguagem do usuário, a especificidade dos descritores, a flexibilidade combinatória dos termos, a solução de ambiguidades ou imprecisões que podem gerar dúvidas quanto ao emprego de um determinado descritor, a estruturação hierárquica e associativa do vocabulário, a disponibilidade de instrumentos auxiliares afins na análise dos termos a serem criados e/ou atualizados.

Dentre os tipos de vocabulários controlados existentes Lancaster (2004, p.19) afirma que os principais são “esquemas de classificação bibliográfica (como a Classificação Decimal de Dewey), listas de cabeçalhos de assuntos e tesouros”, além destes, existem também as ontologias, que não são exatamente um vocabulário controlado, porém podem ser utilizadas para tal finalidade.

Pickler (2014, p. 37) define um tesouro como um instrumento de controle de vocabulário composto por descritores, que são entradas individuais ordenadas segundo as relações recíprocas existentes entre eles. A estrutura de um tesouro é baseada nos descritores reunidos por áreas específicas do conhecimento, que são, por sua vez, subdivididas em categorias principais e depois em grupos.

O uso de ontologias como instrumento de organização da informação vem sendo estudado há algum tempo, principalmente devido a sua característica formal, onde é possível

por exemplo usá-la para representar o conhecimento, tal como um tesouro, diferindo deste no sentido de que as ontologias podem ser criadas para serem utilizadas por sistemas computacionais, permitindo que elas sejam usadas como ferramentas de auxílio aos processos de indexação automática.

O processo de indexação é subjetivo e envolve fatores como os usuários do sistema e o contexto, desta forma pode-se afirmar que esta é uma tarefa complexa de ser formalizada para que seja possível automatizá-la. A adoção de normas e procedimentos padronizados pode promover uma melhora na uniformidade das ações desenvolvidas durante o processo de indexação, sendo que cada instituição é responsável por estabelecer seu conjunto de normas para nortear estes procedimentos, caracterizando assim as chamadas “Políticas de Indexação” (PICKLER, 2014, p.38).

As políticas de indexação fornecem parâmetros para a tomada de decisão em situações dúbias durante a indexação, sendo assim é necessário para sua definição o levantamento de vários aspectos, tais como as necessidades informacionais dos usuários que farão uso do sistema.

Fujita (2012, p.22) define a política de indexação como “um conjunto de procedimentos, materiais, normas e técnicas orientadas por decisões que refletem a prática e princípios teóricos da cultura organizacional de um sistema de informação”.

Quanto a constituição, Cesarino (1985, p. 165) explica que uma política de indexação só pode ser estabelecida depois de observados os seguintes aspectos:

- Identificação das características do usuário (áreas de interesse, nível, experiência, atividades que exercem);
- Volume e características da literatura a ser integrada ao sistema;
- Volume e características das questões propostas pelo usuário;
- Número e qualidade dos recursos humanos envolvidos;
- Determinação dos recursos financeiros disponíveis para criação e manutenção do sistema;
- Determinação dos equipamentos disponíveis.

A utilização de uma política de indexação, segundo Fujita (1999a) não deve ser vista apenas como uma lista de procedimentos a serem seguidos, mas sim como um conjunto de decisões que sejam capazes de esclarecer os interesses e objetivos de um sistema de informação. A autora salienta que a política decide não só sobre a consistência dos procedimentos de indexação em relação aos efeitos que se necessita obter na recuperação, mas, sobretudo, sobre a delimitação de cobertura temática em níveis qualitativos e quantitativos tendo em vista os domínios de assuntos e as demandas dos usuários.

Outro fator importante que torna a adoção de uma política de indexação indispensável é que nela são definidos os parâmetros de exaustividade e especificidade com as quais o sistema vai operar, Pickler (2014) afirma que estes fatores constituem uma decisão política, uma vez que eles afetam diretamente a recuperação de informações.

Na concepção de Lancaster (2004, p.3) a exaustividade é como uma medida de extensão em que todos os assuntos discutidos em certo documento são reconhecidos na operação de indexação e traduzidos para a terminologia do sistema. Já a especificidade é definida por Pickler (2014) como o grau de precisão com que um termo define determinado conceito no documento. A autora afirma que há perda de especificidade quando um conceito é representado por um termo com significado mais genérico, sendo ideal identificar os conceitos com os termos mais específicos quanto possível.

Fujita (2012, p. 26) explica que existe uma reciprocidade de efeitos entre a exaustividade e a especificidade, já que quando a quantidade de termos é considerada pequena ou grande é necessário garantir uma maior cobertura na representação do conteúdo, usando termos mais específicos ou genéricos. Esta etapa de seleção dos termos mais específicos e mais exaustivos pode ser desenvolvida pelo indexador com a utilização de um vocabulário controlado que demonstre os níveis hierárquicos entre os termos identificados no conteúdo. A autora destaca, ainda, que a exaustividade e a especificidade consistem em opções de julgamento do indexador, sob a influência da política de indexação estabelecida pelo sistema de informação e acionadas por um processo cognitivo, durante a representação de conteúdos documentários com termos de indexação que terão efeito direto na recuperação causando precisão e revocação.

Pickler (2014) aponta que além da exaustividade e da especificidade, também podem ser considerados fatores determinantes da qualidade da indexação a correção e a consistência.

A correção na indexação significa a ausência de erros, e isto, na prática, é de suma importância. A indexação pode ter falhas por duas razões: por omissão, quando um termo é omitido e por inclusão, quando há a adição de um termo sem necessidade. Portanto, uma indexação satisfatória depende da relação entre a presença de descritores corretos e a ausência de omissões (SOERGEL, 1994, p. 593).

De acordo com Fujita (2012, p. 28):

a correção como variável individual necessita que o indexador tenha conhecimento prévio não só de habilidades de leitura inatas e construídas ao longo de sua vida, mas também de estratégias profissionais de leitura documentária adquiridas na formação inicial e na formação em serviço que o habilite em leitura documentária para indexação.

Já a consistência é definida por Zunde e Dexter (1969, p. 259) apud Fujita (2012, p. 81) como “[...] o grau de concordância na representação da informação essencial de um documento por meio de um conjunto de termos de indexação selecionados por cada um dos indexadores de um grupo”. Fujita (2012, p.84) afirma que “a inconsistência é uma característica inerente a indexação”, já Pickler (2014, p.41) complementa afirmando que “a subjetividade está presente não só na indexação, mas em cada unidade documentária ou base de dados”.

Após discorrer sobre as políticas de indexação e também sobre o processo em si, cita-se uma fala de Fujita (2012, p.24), onde a autora afirma que “a principal justificativa da indexação é a necessidade de recuperação de informação” no entanto para que seja possível fazer uma indexação adequada é necessário elaborar uma política de indexação, pois esta deve esclarecer e definir variáveis como a especificidade, a exaustividade e a correção, fatores estes que podem contribuir positiva ou negativamente para a etapa de recuperação.

Finda-se aqui a conceituação sobre o processo de indexação, para iniciar a exploração da indexação automática, processo realizado por ferramentas computacionais, cujos objetivos são os mesmos da indexação realizada por humanos, a representação das informações visando sua posterior recuperação.

A indexação automática é um processo realizado por computadores e algoritmos programados para efetuar o processo seleção dos termos mais significativos de um documento, indicando assim seu assunto para que seja possível sua posterior recuperação.

A grande diferença da indexação automática para a indexação realizada por humanos está na objetividade e uniformidade, características inerentes aos sistemas computacionais,

pois, um computador utilizando um software é capaz de tomar as mesmas decisões e adotar as mesmas políticas para todos os documentos que ele processar, independentemente de fatores externos como volume de documentos e conteúdo dos mesmos.

Ferneda (2013) destaca que embora a prática de indexação possa ser regulada por princípios e políticas institucionais, a indexação manual é dependente de critérios subjetivos e pessoais, características inerentes a um indexador humano, tornando-se assim estes fatores não controláveis, podendo assim afetar o resultado da indexação. O autor ainda justifica a necessidade de pesquisas e estudos da indexação automática com base na explosão informacional, na quantidade de documentos publicados e do advento da internet, meio extremamente propício a difusão de informações e documentos.

Em seu livro sobre indexação Keyser (2012) discorre sobre aspectos positivos e negativos da indexação manual, apresentando uma visão geral sobre ambas as formas de indexação. Dentre os argumentos contrários ao processo manual Keyser (2012, p. 40) destaca os seguintes:

- É lenta;
- É custosa, observa-se aqui que o autor se refere a custos financeiros;
- Não é detalhada o suficiente;
- Não necessariamente conduz a uma melhor recuperação;
- Os indexadores humanos podem discordar entre eles quanto aos termos atribuídos;

Quando consideramos que o aumento da produção de conteúdos e documentos é exponencial e não tende a diminuir pode-se concluir que a indexação feita de forma manual é custosa, e não apenas no que se refere ao aspecto financeiro, mas também custosa no que se refere ao tempo dispendido pelos indexadores, ao grau de dificuldade de realização da tarefa além de outros aspectos levantados pelo autor supracitado.

Outro fato a ser destacado é que a qualidade da indexação é diretamente proporcional ao tempo dispensado para a realização da tarefa, ao considerar que a demanda aumenta vertiginosamente, porém a quantidade de bibliotecários e profissionais da indexação não acompanha este crescimento observa-se uma situação onde há a necessidade de redução do tempo gasto ao indexar cada documento, o que pode ocasionar a redução da qualidade da indexação e conseqüentemente da recuperação destes documentos.

Ao discorrer sobre os aspectos favoráveis a indexação manual Keyser (2012, p. 51) destaca algumas falhas críticas da indexação automática, tais como:

- Não proporciona uma visão coerente dos termos de indexação;
- Não resolve os problemas relacionados a sinônimos ou variações ortográficas;
- Não leva em conta o contexto;
- Não permite a navegação entre assuntos relacionados;

Tais dificuldades inerentes à indexação automática são verdadeiras, justificando assim a necessidade de maiores pesquisas e aprimoramentos nos métodos utilizados para se realizar os processos, como a proposta deste trabalho, utilizar ontologias como um recurso de auxílio aos métodos de indexação.

Os problemas relacionados a sinônimos e as variações ortográficas podem ser minimizados em grande parte utilizando uma ontologia, pois é possível declarar um ou mais termos e sinônimos para representar um termo representativo. Isto é feito utilizando a descrição de uma classe pertencente a uma ontologia e é detalhado nas próximas seções.

A navegação entre assuntos relacionados e também pode ser proporcionada com o uso de ontologias, pois adotando estas como um tipo de vocabulário controlado é possível apresentar aos utilizadores de um sistema de recuperação de informações todos os termos utilizados durante a indexação, permitindo aos usuários uma maior familiarização com o vocabulário do sistema em questão.

A questão de contexto é um pouco mais delicada, pois tornar uma máquina capaz de compreender o contexto de determinado documento não é tarefa simples, no entanto, ao analisar o fato de que uma ontologia pertence a determinado domínio pode-se após extrair um termo de um documento busca em todas as ontologias disponíveis em um determinado sistema, fornecendo assim uma pista de qual seria o contexto do documento. Se expandirmos esse processo para um conjunto de termos extraídos do documento, pode-se comparar os domínios de cada termo resultante da busca nas ontologias, a assim criar uma forma de determinar qual o domínio, ou qual o assunto principal de um documento. Tem-se assim uma possível solução para o problema apontado por Keyser.

Após discorrer sobre aspectos contrários e favoráveis acerca do processo de indexação automática, chegamos à conclusão que é possível que a indexação automática atinja níveis de qualidade aceitáveis, porém isso demanda ainda pesquisas e experimentos como este trabalho.

A indexação essencialmente pode ser de dois tipos distintos, definidos por Lancaster (2004, p.18): a indexação por extração e a indexação por atribuição. Na primeira os termos são extraídos do próprio documento, o indexador utilizando suas técnicas e seguindo as políticas de indexação deve selecionar termos apenas dentro do documento e questão, selecionado os mais representativos dentro do contexto do documento. Já na indexação por atribuição utiliza-se um elemento externo ao documento, como um conjunto de termos previamente definidos e normalizados, um vocabulário controlado por exemplo, onde o indexador deverá escolher termos que melhor representam o documento a ser indexado (FERNEDA, 2013, p. 84).

De forma semelhante a indexação manual, Lancaster (2004, p.285) explica que a indexação automática também pode ser feita por extração automática ou atribuição automática. A indexação por extração automática é geralmente implementada utilizando métodos matemáticos de frequência de palavras encontradas no texto, atribuindo ao mesmo os descritores potencialmente mais significativos.

Na indexação por atribuição automática utiliza-se também uma abordagem matemática e/ou estatística para obter os termos potencialmente mais representativos, sendo essa a primeira etapa deste tipo de atribuição. Uma segunda etapa consiste em comparar os descritores obtidos na primeira etapa com algum elemento externo ao texto, visando assim normalizar os termos de indexação atribuídos aos documentos.

3.1 Indexação por extração automática

Neste tipo de método palavras ou expressões do documento são extraídas e utilizadas para representar o conteúdo do texto como um todo. Por meio da frequência, análise do título, resumo, palavras-chave entre outros é possível programar um algoritmo para encontrar os termos supostamente mais representativos em um documento textual.

Lancaster (2004, p.286) afirma que a indexação automática baseada na frequência de palavras tem origem na década de 1950 e no trabalho de Luhn (1957) e Baxendale (1958). O autor ainda explica que os primeiros métodos funcionavam da seguinte forma: um programa era responsável por contar as palavras de um texto, compará-las com uma lista de palavras

proibidas, eliminando assim os termos não-significativos e, em seguida, ordenar as palavras restantes por frequência, sendo as palavras do topo da lista os “melhores termos de indexação”.

Em meados da década de 60, Gerald Salton iniciou o desenvolvimento do Sistema SMART (*System for the Manipulation and Retrieval of Text*), com o qual importantes conceitos e técnicas relacionados à recuperação de informação foram testados e avaliados. Este sistema incorporava um conjunto de ferramentas para extração automática, sendo que o processo é dividido nas seguintes etapas (SALTON; MCGILL, 1983, p.131 *apud* FERNEDA 2013, p. 87):

1. Identificar e isolar cada palavra do texto do documento;
2. Eliminar palavras com grande frequência e pouco valor semântico (stop words) tais como preposições, artigos, etc.;
3. Remover afixos (prefixos e sufixos) das palavras restantes, reduzindo-as a seu radical (processo conhecido como stemming)
4. Incorporar os radicais (termos) ao vetor do documento e atribuir-lhes um peso, calculando por meio da medida $tf \times idf$.

As medidas $tf \times idf$ são definidas por Ferneda (2013, p.87 e 2012, p.37) como formas de se atribuir pesos aos termos de indexação, sendo que tf (*term frequency*) faz referência ao número de vezes que determinado termo aparece no texto de um documento. Isto é útil pois assume-se que a importância de um termo na representação de um documento qualquer é proporcional a sua frequência. No entanto, um termo que aparece em todos os documentos de um corpus é de pouca utilidade, pois possui pouco poder de discriminar um determinado documento. A estatística global que mostra como o termo está distribuído pelo corpus é denominada *inverse document frequency* (idf), sendo que a multiplicação destas medidas é bastante usada para atribuição de pesos aos termos que representam um determinado documento.

Lancaster (2004, p.287) explica o fato de termos que ocorrem com frequência tanto no documento quanto na base de dados não são bons discriminantes com o exemplo:

[...] as palavras biblioteca e informação não seriam bons discriminantes de itens numa coleção de biblioteconomia e ciência da informação. Assim, num documento a palavra biblioteca aparece 12 vezes, enquanto a palavra amianto só ocorre quatro vezes, no entanto, o último termo é muito melhor discriminante, uma vez que se trata de um termo que raramente ocorre na literatura da biblioteconomia. Seria um termo altamente importante numa coleção deste assunto, mesmo que ocorresse uma única vez no documento.

A indexação por extração automática considera que as palavras pertencentes a um documento possuem semântica suficiente para representar os assuntos tratados por ele, porém para selecionar os termos que representarão este documento é necessário realizar uma série de etapas e cálculos matemáticos, como retirar palavras que possuem pouco valor semântico, como artigos, preposições, conjunções, etc. Analisar o qual raro é este termo dentro do corpus em questão, comparar cada um os vocabulários controlados além de outros métodos não abordados, demonstrando assim toda a complexidade deste processo.

3.2 Indexação por atribuição automática

Na indexação por extração automática as etapas envolvidas são basicamente, contar a quantidade de palavras, calcular a frequência no documento e corpus, comparar estas palavras com listas, remover parte das palavras (de acordo com alguma regra), e, em síntese, todas operações que podem ser automatizadas de forma relativamente fácil, visto que computadores e algoritmos realizam tarefas objetivas como essas de forma simples e eficiente.

No processo de indexação por atribuição automática há uma complexidade maior envolvida, pois escolher termos que representem o assunto de um determinado documento requer uma análise do contexto, sendo esta uma operação subjetiva, logo, mais difícil de ser automatizada.

Na indexação por atribuição existe a necessidade de se fazer uma indexação por extração como uma primeira etapa, objetivando obter um conjunto de candidatos a descritores retirados do próprio documento. Em seguida cada um destes descritores deve ser submetido a um vocabulário controlado, visando obter a forma normalizada, ou ainda outros termos relacionados, possibilitando assim a escolha de descritores padronizados, que podem resultar em melhores resultados na recuperação da informação.

Lancaster (2004, p. 289) afirma que a maior parte da indexação manual utiliza a atribuição como metodologia, pois, apontar o assunto um determinado texto após sua leitura é uma tarefa relativamente simples para os humanos, sendo desta forma a opção pela indexação por atribuição mais natural.

O autor supracitado explica que a maneira mais “óbvia” de automatizar a indexação por atribuição é criar para cada termo do vocabulário controlado uma espécie de “perfil” de palavras ou expressões que costumam ocorrer nos documentos aos quais o indexador atribuiria esse

termo. Desta forma seriam necessárias duas etapas distintas, em um primeiro momento seria necessário fazer uma extração de termos do texto por meio de técnicas e estatísticas, a segunda etapa seria comparar cada termo com os perfis já existentes buscando correspondências, obtendo assim os termos adequados para a representação do texto.

Estes perfis criados dentro dos vocabulários controlados serviriam basicamente para que os computadores fossem capazes de identificar qual o assunto dos documentos, atribuindo assim termos que facilitem a recuperação por meio de termos normalizados.

Uma das vantagens do uso de vocabulários controlados destacada por Fernela (2013, p.89) é que estes podem ser disponibilizados aos usuários durante a etapa onde estes criam sua expressão de busca. Fornecendo acesso à terminologia empregada na indexação dos documentos é possível compatibilizar a linguagem dos usuários a utilizada na representação dos documentos, resultando assim em uma recuperação mais eficiente.

Quanto aos vocabulários controlados utilizados durante a indexação por atribuição automática é necessário que estes sejam processáveis por computadores, inviabilizando assim o uso de cabeçalhos de assuntos ou tesouros. Fernela (2013, p.89) propõem que embora uma ontologia não seja originalmente uma ferramenta utilizada recurso de controle terminológico é possível utilizá-las como se comportarem como vocabulários de domínio, permitindo assim realizar inferências e derivações semânticas.

3.3 Indexação automática utilizando ontologias

A utilização de ontologias no processo de indexação por atribuição automática se justifica pela necessidade de um elemento externo que contenha um conjunto de termos para servir aos propósitos da etapa de tradução da indexação, as ontologias destacam-se para esta finalidade pois são artefatos que tratam da organização de objetos, suas propriedades e seus relacionamentos em um determinado domínio de conhecimento (Chandrasekaran; Josephson; Benjamins, 1999).

As ontologias abrem novas perspectivas para o desenvolvimento de sistemas de indexação automática pois oferecem uma estrutura conceitual e terminológica restrita a um domínio e representadas em linguagens processáveis por computador, tal como XML ou OWL, o que permite sua utilização nos mais variados processos computacionais (FERNEDA, 2013 p. 90).

Ainda segundo o autor, a utilização de uma base ontológica possibilita uma abordagem mais rica para a indexação, pois permite oferecer algum nível de análise semântica. Após extrair os termos de um documento é possível mapear estes em uma ontologia, padronizando o vocabulário e contextualizando de forma mais eficiente os termos no domínio da ontologia, solucionando assim possíveis ambiguidades.

O uso de uma ontologia nos processos de indexação e recuperação da informação é apenas um dos possíveis. Outros usos, como os destacados por McGuinness (2003), podem ser organização de taxonomias navegacionais, suporte à navegação entre conteúdos relacionados, verificação de consistência de informações, interoperabilidade entre aplicações e diferentes ontologias entre outros usos.

No próximo capítulo são abordados os conceitos essenciais sobre as ontologias, seus formatos de arquivo, linguagem XML e OWL, sua utilização e posicionamento dentro do bojo da ciência da informação.

4

Ontologias

O termo ontologia possui definições distintas dependendo da área de conhecimento ao qual é citado. Ferneda (2013, p. 48) afirma que na filosofia o termo ontologia possui sua origem na Metafísica, a “Filosofia Primeira” de Aristóteles (384-322 a.C.), que trata do estudo do ser em sua essência.

O Aulete Digital⁴, um dicionário de língua portuguesa na internet define a palavra ontologia de forma bastante abrangente:

(on.to.lo.gi.a)

sf.

1. **Fil.** Parte da filosofia que estuda a natureza dos seres, o ser enquanto ser.
2. **Fil.** Doutrina sobre o ser.
3. **Hist. Med.** Doutrina segundo a qual os fenômenos patológicos têm existência própria, não tendo relação com fenômenos fisiológicos.
4. **Inf.** Campo da informática que trata de conceitualizar de forma explícita e formal (portanto processável por máquina e compartilhável) conceitos e restrições relacionados a certo domínio de interesses

[F.: *ont(o)*- + *-logia*.]

Dentre os significados apresentados, a definição ligada ao campo da informática é a que mais se aproxima dos propósitos almejados por esta pesquisa, pois permite que utilizemos a ontologia como uma ferramenta de auxílio a organização e representação do conhecimento de forma processável por computadores.

No âmbito da ciência da computação uma definição clássica é a de Gruber (1995), para o qual uma ontologia é uma especificação formal e explícita de uma conceitualização

⁴ <http://www.aulete.com.br/ontologia>. Acessado em 28/09/2015

compartilhada. Formal diz respeito a “ser legível por computador”. Explícita, indica que os elementos estão claramente definidos; conceitualização refere-se a um modelo abstrato de um fenômeno; e compartilhada significa que os conceitos presentes representam um conhecimento consensual, aceito por um grupo de pessoas.

Uma ontologia pode ser considerada como um vocabulário de representação, geralmente especializado em um domínio ou assunto, qualificado por conceituações de tipos de objetos e suas relações no mundo. Em outras palavras, é um corpo de conhecimento que descreve algum domínio, utilizando um vocabulário de representação (CHANDRASEKARAN; JOSEPHSON; BENJAMIN, 1999).

Jasper e Uschold (1999, p.11-2) ressaltam a necessidade de explicitação dos relacionados entre os termos pertencentes a uma ontologia:

Uma ontologia pode possuir uma variedade de formas, mas necessariamente incluirá um vocabulário de termos, e alguma especificação de seus significados. Isto inclui definições e uma indicação de como os conceitos estão inter-relacionados, o que impõem uma estrutura no domínio e restringe possíveis interpretações de termos.

Os autores supracitados destacam um ponto crucial do uso de ontologias, a possibilidade de criar restrições, explicitar as relações entre os termos, agregar rótulos em múltiplos idiomas conceituando assim cada termo com expressividade suficiente para que este sejam compreensíveis tanto por humanos quanto por máquinas com o mesmo nível de compreensão.

A utilização de ontologias oferece uma série de vantagens, sendo que no âmbito da ciência da computação destacamos alguns aspectos apontados por Guimarães (2002, p.53):

- Ontologias fornecem um vocabulário para representação do conhecimento. Vocabulário esse que traz uma conceitualização que o sustenta, evitando ambiguidades.
- Ontologias permitem o compartilhamento de conhecimento. Sendo assim, caso exista uma ontologia que modele adequadamente certo domínio de conhecimento, essa pode ser compartilhada e usada por pessoas que desenvolvam aplicações dentro desse domínio.
- Fornecem uma descrição exata do conhecimento. Diferentemente da linguagem natural em que as palavras podem ter semântica totalmente diferente conforme o seu contexto, a ontologia por ser escrita seguindo uma estrutura formal, não deixa

espaço para o gap semântico existente na linguagem natural. Por exemplo, a palavra “globo” pode significar um corpo esférico, porém também é o nome de um canal de televisão brasileiro. A descrição exata de cada termo em uma ontologia evita a interpretação dúbia de termos como este.

- Há possibilidade de mapeamento do idioma da ontologia sem que com isso seja alterada a sua conceitualização, ou seja, uma mesma conceitualização pode ser expressa em vários idiomas.
- É possível estender o uso de uma ontologia genérica de forma a que ela se adéque a um domínio específico.

As vantagens apresentadas por Guimarães são descritas como positivas a área de ciência da computação, porém a ciência da informação e a área de recuperação da informação mais especificamente podem ser fortemente beneficiadas por estes aspectos positivos provindos do uso de ontologias, compreendendo assim a necessidade da aproximação das áreas e o compartilhamento das tecnologias.

Quando ao nível de generalidade Fernela (2013, p.58) afirma que as ontologias podem ser classificadas em três níveis: ontologias genéricas, as que descrevem os conceitos mais gerais, como objetos, ações, eventos, etc. As ontologias de domínio e de tarefa que descrevem respectivamente um domínio como medicina ou direito ou uma tarefa ou atividade como uma venda; por último temos as ontologias de aplicação que descrevem conceitos dependentes de um determinado domínio e tarefa particular.

Uma ontologia é composta por uma série de componentes, uma breve descrição de cada elemento é apresentada por Ramalho (2010, p.38):

- **Classes e Subclasses:** As classes e subclasses de uma ontologia agrupam um conjunto de elementos, “coisas”, do “mundo real”, que são representadas e categorizadas de acordo com suas similaridades, levando-se em consideração um domínio concreto. Os elementos podem representar coisas físicas ou conceituais, desde objetos inanimados até teorias científicas ou correntes teóricas;
- **Propriedades Descritivas:** Descrevem as características, adjetivos e/ou qualidades das classes;

- **Propriedades Relacionais:** Trata-se dos relacionamentos entre classes pertencentes ou não a uma mesma hierarquia, descrevendo e rotulando os tipos de relações existentes no domínio representado;
- **Regras e Axiomas:** Enunciados lógicos que possibilitam impor condições como tipos de valores aceitos, descrevendo formalmente as regras da ontologia e possibilitando a realização de inferências automáticas a partir de informações que não necessariamente foram explicitadas no domínio, mas que podem estar implícitas na estrutura da ontologia;
- **Instâncias:** Indicam os valores das classes e subclasses, constituindo uma representação de objetos ou indivíduos pertencentes ao domínio modelado, de acordo com as características das classes, relacionamentos e restrições definidas;
- **Valores:** Atribuem valores concretos às propriedades descritivas, indicando os formatos e tipos de valores aceitos em cada classe.

A construção de uma ontologia, de acordo com FERNEDA (2013, p.62) pode ser pensada como:

[...] a união de peças que formam uma estrutura completa. Classes e subclasses definem um “esqueleto” na forma de uma hierarquia [...], complementada por propriedades descritivas, propriedades relacionais, regras e axiomas. A sua abrangência (domínio) deve ser previamente definida e estabelece uma área do conhecimento ou uma parte do mundo que se pretende tratar.

Conforme o contexto apresentado nota-se que as ontologias fornecem uma estrutura conceitual que visa representar formalmente os conceitos e suas relações, regras e restrições lógicas de um determinado domínio, cujo objetivo principal de sua construção é a necessidade de um vocabulário compartilhado cujas informações possam ser trocadas e reusadas por usuários ou agentes inteligentes (FERNEDA, 2013; SANTARÉM SEGUNDO, 2010).

4.1 Ontologias e a Ciência da Informação

A Ciência da Informação é uma área ligada as tecnologias de informação desde suas origens, como coloca Marcondes e Campos (2008), os autores afirmam que a área de CI vem de uma longa tradição teórica, metodológica e prática, que converge para questões mais atuais ligadas a Web Semântica e a construção e utilização de ontologias viabilizando assim o acesso

e a organização de acervos documentais assim como o processamento semântico de informações em computadores.

Segundo Soergel (1999) e Vickery (1997), o termo ontologia começou a ser utilizado na literatura da Ciência da Informação no final da década de 1990, mas principalmente por pesquisadores da área de Organização do Conhecimento. Nesta época, os instrumentos e métodos de classificação despertavam maior interesse da Computação, devido principalmente a necessidade de desenvolvimento de instrumentos de organização da informação no ambiente Web.

A Organização do Conhecimento vem se consolidando como um importante campo de investigação da Ciência da Informação, a partir da fundação da *International Society for Knowledge Organization* (ISKO), em 1989 (PICKLER, 2014, p.60). A autora ainda explica que:

A Organização do Conhecimento é uma disciplina que estuda uma série de fundamentos teóricos do tratamento e recuperação da informação, avaliando o uso de instrumentos lógico-linguísticos para controlar os processos de representação, classificação, ordenação e armazenamento do conteúdo informativo dos documentos com a finalidade de permitir sua recuperação e disseminação.

A organização do conhecimento visando seu posterior uso é de importância primordial, desde a antiguidade até os dias atuais. Marcondes e Campos (2008, p.109) pontuam que o conhecimento científico, tão importante para a sociedade, vem passando por um desafio com o surgimento da Web. A velocidade em que são transferidos documentos e informações pela internet é indiscutível, um avanço grandioso, o que acabou permitindo um crescimento desordenado dos conteúdos disponíveis na rede, e o mais crítico é que grande parte do volume de materiais na rede são compreensíveis apenas por seres humanos.

As ontologias então apresentam-se como:

Uma categoria de instrumentos de representação do conhecimento, possibilitando e favorecendo melhorias nos processos de representação, organização, disseminação e recuperação de conteúdos documentais. Se na Computação uma ontologia pode ser considerada um artefato tecnológico, no contexto da Ciência da Informação caracteriza-se como um instrumento de nível epistemológico, concebido para favorecer a representação formal dos conceitos e dos relacionamentos existentes entre eles em um domínio específico (FERNEDA, 2013, p.64)

Entre os instrumentos de representação tradicionalmente utilizados na área de Ciência da Informação, os tesouros são os mais semelhantes às ontologias. Como destacam Sales e Café (2008), ambos os instrumentos são constituídos por linguagens de estruturas combinatórias, de

caráter especializado, representando termos e conceitos organizados a partir de tipos de relacionamentos, no entanto, estudos têm constatado que, apesar de possuírem características em comum, tais instrumentos constituem-se como diferentes modelos de representação do conhecimento.

Outro fator determinante para a distinção entre ontologias e os demais modelos de representação utilizados na CI é a natureza das relações. Pickler (2014, p.61) destaca que para possibilitar a realização de inferências automáticas é necessário que as relações expressas na ontologia sejam formalmente descritas, rotuladas e categorizadas. Assim as ontologias proporcionam liberdade para representar tipos de relacionamentos que não seriam possíveis em outros modelos de representação.

O potencial de representação do conhecimento, da expressividade alcançada com o uso de ontologias as colocam como um instrumento valioso ao arsenal teórico e prático da Ciência da Informação, mas, para isso, é necessário que os profissionais da informação sejam capazes de utilizar os recursos oferecidos pelas mesmas, sendo necessários estudos como este, que se presta a colocar em prática as ontologias como instrumento de representação do conhecimento, buscando validar as propostas e hipóteses que apontam para melhores resultados utilizando ontologias.

Este estudo busca estreitar a relação entre as Ciências da Informação e Computação, pois se presta a utilizar ferramentas e softwares para verificar os resultados de uma indexação automática apoiada na utilização de ontologias como ferramentas de apoio a decisão em situações dúbias durante a indexação automática de documentos textuais.

4.2 Web Ontology Language (OWL)

Atualmente a OWL é recomendada pelo consórcio W3C como a principal linguagem para a construção de ontologias. Ela tem como objetivo principal atender às necessidades de aplicação da Web Semântica e ser efetivamente utilizada por aplicações que necessitem processar o conteúdo de informações, e não somente apresentar a visualização destas informações. (SANTARÉM SEGUNDO, 2010, p.127).

A documentação oficial disponibilizada pelo W3C afirma que a OWL é uma linguagem de web semântica projetada para representar o conhecimento em toda sua riqueza e complexidade, além de grupos de coisas e as relações entre as coisas. Ela é baseada em lógica

computacional onde o conhecimento é expresso de forma que o computador possa explorá-lo e verificá-lo sistematicamente, além de ter acesso a informações explícitas ou implícitas por meio de inferências. A OWL é parte da “pilha” de tecnologias propostas pelo W3C, que inclui ainda RDF, RDFS, SPARQL, entre outras.

A linguagem OWL é subdividida em três sublinguagens, projetadas para uso de implementadores e comunidades específicas: OWL Lite, OWL DL e OWL Full, seus detalhes e características são apresentadas na sequência:

- OWL Lite: dá suporte à criação de hierarquias simplificadas, com restrições simples. Devido a menor conjunto de funcionalidades ela se torna mais simples e de fácil utilização, sendo um de seus principais objetivos fornecer um rápido caminho de migração para tesauros e outras taxonomias.
- OWL DL: possui o mesmo vocabulário da linguagem OWL Full e dá suporte aos usuários que desejam o máximo de expressividade, sem perdera completude computacional (todas as conclusões são garantidas de serem computadas) e capacidade de decisão (todas as computações serão finalizadas em um tempo finito) dos sistemas de raciocínio. O OWL DL inclui todos os construtores da linguagem OWL com restrições, como separação entre tipos (uma classe não pode ser ao mesmo tempo um indivíduo ou tipo, e uma propriedade não pode ser ao mesmo tempo um indivíduo ou uma classe). OWL DL tem expressividade menor que o OWL Full, mas conta com melhor eficiência, pois garante que todas as conclusões sejam computáveis e resolvidas em tempo finito. Ela tem este nome devido a sua correspondência à Lógica de Descrição ou Description Logic.
- OWL Full é a versão com o máximo de expressividade e liberdade sintática do RDF, sem nenhuma garantia computacional. A linguagem OWL Full não conta com as restrições da OWL DL, e justamente por isso pode ser bem mais adaptada a situações onde o ponto mais importante é a expressividade. Ela suporta o mesmo subconjunto da OWL DL porém tem restrições diferentes, como, por exemplo, permitir que uma classe seja ao mesmo tempo uma classe e um indivíduo.

A escolha de uma ou outra especificação da linguagem está diretamente ligada a análise prévia do domínio e do tipo de ontologia que se pretende criar (SANTARÉM SEGUNDO, 2010, p.129).

Os desenvolvedores de ontologias devem escolher a versão de acordo com suas necessidades. Basicamente a escolha consiste em balancear o nível de expressividade com decidibilidade e eficiência na inferência de informações, isto porque, quanto maior o nível de expressividade utilizado na descrição de uma ontologia, maior será a complexidade dos algoritmos envolvidos em processos de busca de informações (FILHO & LÓSCIO, 2009).

Um arquivo OWL é formado por alguns “blocos” bem definidos, sendo que cada um deles tem um papel muito claro junto a estrutura da ontologia. Agora faremos uma breve descrição de cada um destes blocos que compõem a estrutura de um arquivo OWL bem formado.

4.2.1 Namespaces

O início de um arquivo OWL é formado pela declaração de *namespaces*, que são os vocabulários que serão empregados dentro da ontologia. Para que os termos dentro de uma ontologia possam ser descritos de forma a serem interpretados sem ambiguidades é de suma importância declarar corretamente os *namespaces*.

O componente padrão inicial de uma ontologia inclui um conjunto de declarações em XML dos *namespaces* incluídas dentro de uma *tag rdf:RDF*. Veja na Figura 2 uma típica declaração dos *namespaces* em uma ontologia sobre vinhos.

Figura 2 – Exemplo de declaração dos *namespaces* em uma ontologia OWL

```
1 <rdf:RDF
2   xmlns      ="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#"
3   xmlns:vin  ="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#"
4   xml:base   ="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#"
5   xmlns:food ="http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#"
6   xmlns:owl  ="http://www.w3.org/2002/07/owl#"
7   xmlns:rdf  ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
8   xmlns:rdfs ="http://www.w3.org/2000/01/rdf-schema#"
9   xmlns:xsd  ="http://www.w3.org/2001/XMLSchema#">
```

Fonte: Adaptado de PICKLER, 2014, p.78

Sobre a Figura 2 é possível observar que a linha 2 não possui prefixo indicando que nessa ontologia todos os nomes sem prefixo terão como domínio a própria ontologia. A linha 3 define o prefixo “vin” para referenciar uma ontologia de vinhos pré-definida. Essa ontologia sobre vinhos é bastante utilizada nos documentos disponibilizados pelo W3C para exemplificar conceitos da Web Semântica e nos diversos manuais sobre a linguagem OWL.

A linha 4 define o prefixo *base*, indicando que essa ontologia se utilizará de uma ontologia já construída anteriormente como base para a construção da nova ontologia. Verifica-se assim a capacidade de reuso da informação existente, pois esta é uma forma de utilizar ontologias prontas para criar novas, mais específicas ou que atendam as novas necessidades.

A linha 5 define o prefixo *food* que se refere uma segunda ontologia já pronta, é desta forma que podem ser utilizadas várias ontologias (e vocabulários) para se representar o conhecimento.

As últimas linhas, 6 até 9 são os prefixos “owl”, “rdf”, “rdfs” e “xsd”, estes são obrigatórios para o uso da linguagem OWL, pois indicam o uso do léxico e da sintaxe referenciados na URLs.

A utilização destes prefixos tem como principal objetivo facilitar o desenvolvimento da ontologia, abreviando a declaração completa das definições, resultando assim em uma maior clareza no código. É possível também utilizar as URLs completas diretamente ao invés dos prefixos, porém isso pode gerar mais dificuldade na leitura e desenvolvimento das ontologias.

4.2.2 Cabeçalhos

Após a definição dos *namespaces*, o próximo passo na criação de um arquivo OWL é a definição de um cabeçalho, que é responsável por indicar um conjunto de informações a respeito da ontologia que está sendo desenvolvida.

É neste momento que devem ser inseridas as informações que dão suporte a tarefas cruciais do desenvolvimento de uma ontologia, como comentários, versão da ontologia, importação de código pré-existente, além da caracterização dos metadados referentes a ontologia que está sendo desenvolvida.

Estas informações devem ser agrupadas dentro da *tag owl:Ontology*, como pode ser visto na Figura 3.

Figura 3 – Exemplo de declaração do cabeçalho de uma ontologia OWL

```
1 <owl:Ontology rdf:about="">
2   <rdfs:label>
3     Ontologia sobre Vinhos
4   </rdfs:label>
5   <rdfs:comment>
6     Um exemplo de ontologia OWL
7   </rdfs:comment>
8   <owl:versionInfo>
9     06/05/2006 15:45:00
10  </owl:versionInfo>
11  <owl:priorVersion rdf:resource="http://www.w3.org/TR/2003/PR-owl-guide-20031215/wine"/>
12  <owl:imports rdf:resource="http://www.w3.org/TR/2004/REC-owl-guide-20040210/food"/>
13
```

Fonte: Adaptado de PICKLER, 2014, p.79

A tag inicial *owl:Ontology* na linha 1 indica que neste bloco estão descritos os metadados da ontologia que está sendo desenvolvida.

A tag *rdfs:label* na linha 2 tem a função de nomear a ontologia. O atributo *rdfs:comment* na linha 5 permite a definição de comentários sobre a ontologia. O atributo *owl:versionInfo* da linha 8 permite inserir um texto contendo informações sobre a versão da ontologia. O atributo *owl:priorVersion* na linha 11 faz uma referência explícita à versão anterior a essa ontologia. O atributo *owl:imports* na linha 12 permite a inserção de dados de outros arquivos dentro do documento que está sendo desenvolvido.

É importante ressaltar a diferença entre indicar o *namespace* para uma ontologia e a importação da mesma usando a tag *owl:imports*. A indicação do *namespace* ocorre quando se deseja utilizar parte da estrutura de outro documento, como definição de classes ou atributos, por exemplo. Já a utilização da importação com a tag *owl:imports* indica que o conteúdo completo da outra ontologia será inserido no documento.

Após descrever um pouco sobre as declarações iniciais, do cabeçalho e dos *namespaces* seguimos abordando os elementos básicos da construção de uma ontologia OWL, são eles as classes, as instâncias destas classes (indivíduos) e as propriedades (relacionamentos) entre essas instâncias.

4.2.3 Classes

As classes proporcionam um mecanismo de abstração para agrupar recursos com características similares, ou seja, uma classe define um grupo de indivíduos que compartilham algumas propriedades (CARVALHO; LIMA, 2005).

As classes são utilizadas para definir um conceito de um determinado domínio e pessoas, automóveis, ou qualquer outra entidade concreta ou abstrata que se deseja representar. As classes e seus relacionamentos hierárquicos são os elementos básicos para a criação da taxonomia que serve de alicerce para a construção de qualquer ontologia. Na linguagem OWL, a classe principal é expressa como *owl:Thing*, sendo que essa classe predefinida é a classe mais genérica (superclasse) da qual derivam todas as demais classes criadas pelo usuário.

Uma classe OWL é definida utilizando a tag *owl:Class* seguida de um atributo identificador *rdf:ID*. A Figura 4 mostra a criação de uma classe.

Figura 4 – Exemplo de definição de uma classe

```
1  
2 <owl:Class rdf:ID="Computador" />  
3
```

Fonte: Adaptado de PICKLER, 2014, p.80

A sintaxe que pode ser vista na Figura 4 é uma das formas mais simplistas de criação de uma classe. Esta classe é referenciada no corpo da ontologia utilizando o caractere “#” seguido do nome da classe, tal como outras linguagens de programação. Assim, para se referir a esta classe podemos utilizar a seguinte declaração: **#Computador**. Isso pode ser usado por exemplo ao se referir a um recurso, usando o código *rdf:resource="#Computador"* ou ainda utilizando a cláusula *about*, como em *rdf:about="#Computador"*.

Uma forma mais comum de se criar classe é criando um bloco delimitado por abertura e fechamento da tag *owl:Class*, como podemos ver na Figura 5. Nesta segunda forma podemos agregar dentro da classe propriedades e relacionamentos desta classe.

Figura 5 – Exemplo de definição de uma classe usando blocos OWL

```
1  
2 <owl:Class rdf:ID="Fabricante">  
3   <rdfs:label>Fabricante</rdfs:label>  
4   ...  
5 </owl:Class>  
6
```

Fonte: Adaptado de PICKLER, 2014, p.81

A Figura 5 mostra a definição de um rótulo usando a propriedade *rdf:label* para a classe Fabricante. Essa é apenas uma das propriedades que pode ser definidas para uma classe OWL.

Como já foi dito uma ontologia deve se apoiar em uma estrutura taxonômica, na qual as classes se organizam em uma forma hierárquica. O código para se declarar isso em uma ontologia OWL é baseado na tag *rdfs:subClassOf*, como podemos ver na Figura 6.

Figura 6 – Exemplo de definição de uma Hierarquia de classes em OWL

```
1
2 <owl:Class rdf:ID="Computador"/>
3   <rdfs:label>Computador</rdfs:label>
4 </owl:Class>
5
6 <owl:Class rdf:ID="Desktop">
7   <rdfs:label>Desktop</rdfs:label>
8   <rdfs:subClassOf rdf:resource="#Computador"/>
9 </owl:Class>
10
11 <owl:Class rdf:ID="Notebook">
12   <rdfs:label>Notebook</rdfs:label>
13   <rdfs:subClassOf rdf:resource="#Computador"/>
14 </owl:Class>
15
16 <owl:Class rdf:ID="AllInOne">
17   <rdfs:label>All in One</rdfs:label>
18   <rdfs:subClassOf rdf:resource="#Desktop"/>
19 </owl:Class>
20
21 <owl:Class rdf:ID="Netbook">
22   <rdfs:label>Netbook</rdfs:label>
23   <rdfs:subClassOf rdf:resource="#Notebook"/>
24 </owl:Class>
25
```

Fonte: Adaptado de PICKLER, 2014, p.82

O código apresentado na Figura 6 mostra que as classes “Desktop” e “Notebook” são subclasses da classe “Computador”. Assim, o conjunto de indivíduos da classe Notebook e os indivíduos da classe Desktop devem ser um subconjunto de indivíduos da classe Computador, isto é, a classe Computador possui características (atributos) que são comuns à classe Notebook e à classe Desktop.

Utilizando a tag *rdfs:subClassOf* é possível definir na ontologia os relacionamentos de especificidade e generalização, considerando o caminho que vai de uma classe para uma subclasse (especificidade) ou de uma subclasse à sua correspondente classe superior (generalização).

Este recurso abre valiosas possibilidades para o processo de recuperação da informação, visto que com essa possibilidade de “navegação” em uma ontologia é bastante útil, por exemplo, é possível a partir de do termo “AllInOne” o sistema de recuperação compreenda que *AllInOne*

é um tipo de Desktop, que por sua vez é um tipo de computador, “compreendendo” assim o contexto da busca de um usuário.

Um outro recurso da linguagem OWL é a definição de classes equivalentes ou sinônimas, para isso é utilizada a tag *owl:equivalentClass*. Na Figura 7 pode-se observar a definição de uma classe equivalente.

Figura 7 – Exemplo de definição de uma classe equivalente

```
1
2 <owl:Class rdf:ID="Laptop"/>
3   <rdfs:label>Laptop</rdfs:label>
4   <owl:equivalentClass rdf:resource="#Notebook"/>
5 </owl:Class>
6
```

Fonte: Adaptado de PICKLER, 2014, p.83

Na Figura 7 uma classe “Laptop” foi definida como uma classe equivalente à classe “Notebook”, desta forma é possível reduzir ambiguidades e agregar conceitos por seus significados verdadeiros, não apenas pela grafia do termo.

Este recurso também pode ser utilizado para promover a interoperabilidade entre duas ontologias diferentes, declarando algumas classes equivalentes é possível estabelecer um consenso entre termos com grafias diferentes e significados idênticos.

Ainda falando sobre a definição das relações em uma hierarquia de classes temos um recurso que permite especificar quando um indivíduo não pode ser membro de duas classes, este recurso é conhecido como disjunção. Na Figura 8 é apresentado o código que explicita que a classe “Desktop” é disjunta à classe “Notebook”, isto é, não pode existir um computador que seja ao mesmo tempo um Desktop e um Notebook.

Figura 8 – Exemplo de definição de uma classe disjunta

```
1
2 <owl:Class rdf:ID="Desktop"/>
3   <rdfs:label>Desktop</rdfs:label>
4   <owl:disjointWith rdf:resource="#Notebook"/>
5 </owl:Class>
6
```

Fonte: Adaptado de PICKLER, 2014, p.84

Filho e Lóscio (2009) destacam que a linguagem OWL permite ainda a criação de novas classes a partir de classes previamente existentes, sendo possível especificar que:

- *unionOf*: é possível criar uma classe como sendo a união de duas ou mais classes. Exemplo: a classe de acadêmicos pode ser definida como a união das classes docentes e discentes.
- *intersectionOf*: uma classe também pode ser originada da intersecção de duas classes existentes. Por exemplo: pode-se criar uma classe “Alunos professores” através da intersecção das classes discentes e docentes.

4.2.4 Indivíduos

Os indivíduos são definidos como objetos do mundo que sempre são ligados às classes previamente definidas, ou seja, são membros das classes. Em uma ontologia todo indivíduo deve pertencer a uma classe existente, que foi criada previamente.

Para inserir um indivíduo em uma ontologia OWL devemos apresentá-lo como membro de uma classe, para isso utiliza-se o código mostrado na Figura 9.

Figura 9 – Exemplo de definição de um indivíduo

```

1
2 <Notebook rdf:ID="MeuComputador" />
3

```

Fonte: Adaptado de PICKLER, 2014, p.87

A Figura 9 indica que o indivíduo “MeuComputador” é uma instância da classe Notebook, definida na Figura 6. Esta construção declara um fato na ontologia que permite ao motor de inferência chegar à seguinte conclusão “MeuComputador é um Notebook”.

Outra forma de criar um indivíduo é defini-lo inicialmente como uma instância da classe geral *owl:Thing* e em seguida defini-lo com um tipo *rdf:type* da classe à qual esse indivíduo está associado. Nesta segunda abordagem a leitura e compreensão da declaração do indivíduo e de suas características ficam mais clara. Veja um exemplo desta forma alternativa para declaração de um indivíduo na Figura 10.

Figura 10 – Exemplo de uma forma alternativa para a definição de um indivíduo

```
1
2 <owl:Thing rdf:ID="MeuComputador" />
3
4 <owl:Thing rdf:about="#MeuComputador">
5   <rdfs:label>Meu Computador</rdfs:label>
6   <rdf:type rdf:resource="#Notebook" />
7 </owl:Thing>
8
```

Fonte: Adaptado de PICKLER, 2014, p.87

4.2.5 Propriedades

Propriedades são recursos da linguagem OWL que têm o propósito de descrever fatos em geral, elas são usadas para estabelecer relacionamentos entre dois indivíduos ou entre indivíduos e valores de dados. Através das propriedades, pode-se fazer referência a todos os membros de uma classe, ou seja, afirmar fatos gerais sobre todos os membros de uma classe ou apenas sobre um indivíduo específico.

A linguagem OWL dispõem de duas categorias principais de propriedades:

- Propriedades de objetos: estabelecem relação entre classes ou indivíduos;
- Propriedades de dados: indicam a relação entre indivíduos e valores de dados, estes valores devem ser expressos em RDF utilizando os tipos definidos no XML Schema

Qualquer propriedade OWL é subclasse da classe *rdf:Property*. Propriedades de objetos são definidas como instância da classe *owl:ObjectProperty*, já as propriedades de dados são definidas como instâncias da classe *owl:DatatypeProperty*. Na Figura 11 pode-se ver a definição de uma propriedade de objeto denominada “Marca” que está associada à classe “Computador”.

Figura 11 – Exemplo de propriedade de objetos

```
1
2 <owl:ObjectProperty rdf:ID="Marca">
3   <rdfs:label>Marca</rdfs:label>
4   <rdfs:domain rdf:resource="#Computador"/>
5   <rdfs:range rdf:resource="#Fabricante"/>
6 </owl:ObjectProperty>
7
```

Fonte: Adaptado de PICKLER, 2014, p.84

Na Figura 11 temos um código que apresenta a definição de uma propriedade de objetos, onde a linha 4 especifica que a classe “Computador” possui uma propriedade denominada “Marca” que deve ser obrigatoriamente preenchida com valores da classe “Fabricante” (linha 5), sendo que esta deve estar previamente definida na ontologia.

O uso de propriedades de dados é demonstrado na Figura 12, onde duas propriedades são definidas, vejamos:

Figura 12 – Exemplo de propriedade de dados

```
1
2 <owl:DatatypeProperty rdf:ID="MemoriaRam">
3   <rdfs:label>Memória RAM</rdfs:label>
4   <rdfs:domain rdf:resource="#Computador"/>
5   <rdfs:range rdf:resource="&xsd;positiveInteger"/>
6 </owl:DatatypeProperty>
7
8 <owl:DatatypeProperty rdf:ID="Nome">
9   <rdfs:label>Nome</rdfs:label>
10  <rdfs:domain rdf:resource="#Fabricante"/>
11  <rdfs:range rdf:resource="&xsd:string"/>
12 </owl:DatatypeProperty>
13
```

Fonte: Adaptado de PICKLER, 2014, p.85

A primeira propriedade apresentada na Figura 12 é uma propriedade da classe Computador, isso é indicado na linha 4, o nome desta propriedade é “MemoriaRam” como pode ser visto na linha 2, ela ainda possui um rótulo em formato mais legível na linha 3 e os valores aceitáveis desta propriedade são inteiros positivos, tal restrição foi especificada na linha 5, por meio da referência ao léxico do XML Schema Datatypes. É possível usar os tipos pertencentes ao XML Schema devido a declaração do *namespace* correspondente (xsd) na Figura 2.

A segunda propriedade da Figura 12 define um nome do tipo *string* para a classe “Fabricante”, seguindo a mesma lógica da propriedade “MemoriaRam”. Além da definição de propriedades como estas supracitadas é possível também criar hierarquias de propriedades, assim como as hierarquias de classes.

Para definir uma hierarquia de propriedades utilizamos a declaração *rdfs:subPropertyOf*, desta forma é possível afirmar que uma determinada propriedade é uma sub-propriedade de alguma outra. Veja na Figura 13 algumas sub-propriedades para a classe Fabricante.

Figura 13 – Exemplo de definição de uma hierarquia de propriedades

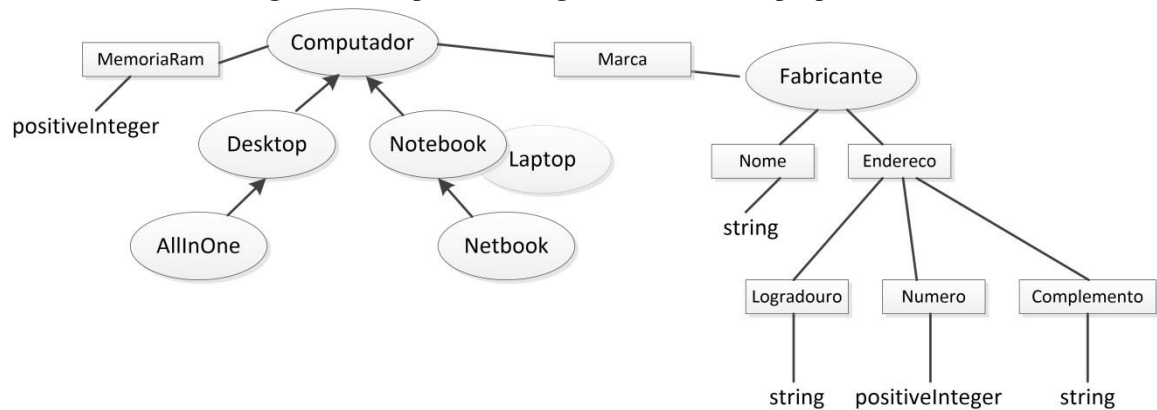
```
1
2 <owl:DatatypeProperty rdf:ID="Endereco">
3   <rdfs:label>Endereço</rdfs:label>
4   <rdfs:domain rdf:resource="#Fabricante"/>
5 </owl:DatatypeProperty>
6
7 <owl:DatatypeProperty rdf:ID="Logradouro">
8   <rdfs:label>Logradouro</rdfs:label>
9   <rdfs:subPropertyOf rdf:resource="#Endereco"/>
10  <rdfs:domain rdf:resource="#Fabricante"/>
11  <rdfs:range rdf:resource="#xsd:string"/>
12 </owl:DatatypeProperty>
13
14 <owl:DatatypeProperty rdf:ID="Numero">
15   <rdfs:label>Número</rdfs:label>
16   <rdfs:subPropertyOf rdf:resource="#Endereco"/>
17   <rdfs:domain rdf:resource="#Fabricante"/>
18   <rdfs:range rdf:resource="#xsd:positiveInteger"/>
19 </owl:DatatypeProperty>
20
21 <owl:DatatypeProperty rdf:ID="Complemento">
22   <rdfs:label>Complemento</rdfs:label>
23   <rdfs:subPropertyOf rdf:resource="#Endereco"/>
24   <rdfs:domain rdf:resource="#Fabricante"/>
25   <rdfs:range rdf:resource="#xsd:string"/>
26 </owl:DatatypeProperty>
27
```

Fonte: Adaptado de PICKLER, 2014, p.86

A Figura 13 apresenta uma propriedade “Endereco” com suas sub-propriedades “Logradouro”, “Numero” e “Complemento”, sendo a propriedade “Endereco” relacionada à classe “Fabricante”.

Uma representação gráfica da hierarquia de classes, junto com as propriedades definidas nos exemplos anteriores é mostrada na Figura 14.

Figura 14 – Representação gráfica de classes e propriedades



Fonte: Adaptado de PICKLER, 2014, p.86

Há ainda uma série de características e especializações que podem ser aplicadas às propriedades, porém estas não serão exemplificadas e detalhadas pois este não é o foco do trabalho, estas propriedades serão citadas de forma breve apenas para demonstrar a capacidade de representação e de expressividade da linguagem OWL.

Filho e Lóscio (2009) apresentam as seguintes especializações de propriedades:

- *TransitiveProperty*: característica que permite especificar quando uma propriedade é transitiva. Exemplo: A relação de subordinação entre funcionários pode ser modelada como uma propriedade transitiva, pois se um funcionário A é subordinado a um funcionário B e este por sua vez é subordinado a um funcionário C, então infere-se que o funcionário A é subordinado ao funcionário C.
- *SymmetricProperty*: duas ou mais propriedades podem ser declaradas simétricas por meio desta especificação. Exemplo: O matrimônio entre duas pessoas pode ser representado por uma propriedade simétrica, pois, se uma pessoa de nome Gabriel é casada com outra de nome Bárbara, a recíproca será verdadeira.
- *FunctionalProperty*: dado um sujeito qualquer, uma propriedade funcional obriga a existência de uma restrição de unicidade para o valor do objeto. É equivalente a afirmar que a propriedade tem restrição de mínima cardinalidade igual a 0 e máxima cardinalidade igual a 1. Exemplo: um curso de graduação é chefiado por apenas um coordenador.
- *minCardinality*: uma propriedade pode ter uma cardinalidade mínima, sendo que se esta for igual a 0 explicita-se que esta propriedade não é obrigatória para uma classe.

Afirmar que a cardinalidade mínima é igual a 1 significa que esta propriedade é obrigatória. Por exemplo, disciplinas têm, no mínimo um crédito.

- *maxCardinality*: assim como é possível especificar uma cardinalidade mínima, é possível definir uma cardinalidade máxima. Definir este valor como 1 significa afirmar que só é permitido no máximo um elemento desta propriedade para uma determinada classe. Um exemplo seria a propriedade “Mãe biológica” que deve possuir cardinalidade máxima para um indivíduo qualquer.
- *cardinality*: quando é preciso informar a cardinalidade mínima e máxima iguais, pode-se usar apenas a propriedade *cardinality*, que é usada para definir quando uma propriedade deve ter exatamente um valor qualquer. Exemplo: professores efetivos de uma universidade devem ter uma carga horária de exatamente 40 horas semanais.

4.2.6 Identificadores e *Labels*

Como foi discutido nesta seção as ontologias podem ser utilizadas para representar o conhecimento sobre um determinado domínio, por meio da descrição dos conceitos e relacionamentos envolvidos neste domínio (PICKLER, 2014, p.87).

Na área da informática e programação é relativamente comum a limitação do uso de caracteres especiais, acentos e espaços. Em algumas situações não é permitido o uso de tais constructos, o que em idiomas como o inglês não é um grande problema, porém em línguas como o português ou o francês que possuem muitas palavras acentuadas é uma limitação.

A linguagem OWL segue a mesma linha de raciocínio, sendo que na declaração dos identificadores (ID) não é permitido o uso de espaços, caracteres especiais, incluindo as letras acentuadas e o uso do “ç”. Logo, em muitas situações não é possível identificar uma determinada entidade da ontologia com os termos corretamente grafados em línguas como o português do Brasil.

Manaf, Bechhofer e Stevens (2010) apresentam um estudo utilizando um corpus contendo 306 ontologias OWL disponíveis na Web. A partir desse corpus identificou-se um conjunto de diferentes estilos utilizados na definição de identificadores de classes, indivíduos e propriedades dessas ontologias (Tabela 1).

Tabela 1 – Estilos de Identificadores em Ontologias

Estilo	Exemplo
Camel Case	IndexacaoBaseadaEmOntologia
Underscore	Indexacao baseada em ontologia
Hífen	Indexacao-baseada-em-ontologia
Camel Case + Underscore	IndexacaoBaseada em ontologia
Camel Case + Hífen	IndexacaoBaseada-em-ontologia
Hífen + Underscore	Indexacao Baseada-em-ontologia
Palavra única	Indexacao

Nesse trabalho, a maioria das ontologias estudadas utilizavam identificadores definidos em algum desses estilos. Porém, há de se considerar que as ontologias utilizadas na pesquisa tinham como base a língua inglesa, que possui certa facilidade na interpretação automática dos identificadores. Por exemplo: o identificador “AutomaticIndexing” (*camel case*) ou “Automatic_indexing” (*underscore*) podem ser facilmente interpretados de forma automática como representantes do conceito “Automatic Indexing”. Porém, em idiomas cujo léxico utiliza acentuação, essa interpretação será mais difícil (PICKLER, 2014, p.88)

Uma forma de contornar este problema é utilizando um recurso da linguagem OWL, a propriedade *label*, com o qual é possível definir de forma precisa o termo que identifica uma entidade (classe, indivíduo ou propriedade), possibilitando a utilização de todo o léxico, incluindo acentos, espaços e caracteres especiais.

A Figura 15 apresenta uma forma bastante simples de utilização da propriedade *label*:

Figura 15 – Exemplo da forma simplificada de utilização da propriedade *label*

```

1
2 <owl:Class rdf:ID="IndexacaoAutomatica">
3   <rdfs:label> Indexação Automática </rdfs:label>
4   ...
5 </owl:Class>
6

```

Fonte: Adaptado de PICKLER, 2014, p.88

A utilização da propriedade *label* permite descrever de forma exata o nome ou rótulo de uma entidade, provendo exatidão independente da língua em que está construída a ontologia. No entanto, é importante destacar que a utilização desta propriedade é opcional, facultando aos criadores da ontologia informar estes rótulos ou não. É permitido também declarar um ou mais *labels* possibilitando assim uma maior expressividade em casos de vários termos se referirem a uma mesma entidade.

Além da declaração mais simples apresentada na Figura 15, é possível utilizar a propriedade *label* com a utilização de um parâmetro *xml:lang* que agrega a informação de qual o idioma daquele rótulo. Desta forma é possível declarar em uma entidade qualquer, vários *labels* cada em seu idioma criando assim a possibilidade de criar ontologias multilíngues.

A Figura 21 demonstra a utilização da propriedade *label* com parâmetros de idioma para cada rótulo.

Figura 16 – Exemplo da utilização da propriedade *label* com os parâmetros de idioma

```
1
2 <owl:Class rdf:ID="IndexacaoAutomatica">
3   <rdfs:label xml:lang="pt">Indexação Automática</rdfs:label>
4   <rdfs:label xml:lang="sp">Indexación Automática</rdfs:label>
5   <rdfs:label xml:lang="en">Automatic Indexing</rdfs:label>
6   <rdfs:label xml:lang="fr">Indexation Automatique</rdfs:label>
7   ...
8 </owl:Class>
9
```

Fonte: Adaptado de PICKLER, 2014, p.89

O uso da propriedade *label* mesmo não sendo obrigatório perante a sintaxe da linguagem OWL pode ser bastante significativo do ponto de vista da recuperação da informação, pois ele garante maior precisão da descrição dos nomes das entidades em idiomas que utilizam acentos em seu léxico, como o português do Brasil. Além disso, temos ainda a possibilidade de reuso de ontologias existentes, pois ontologias elaboradas em qualquer idioma podem ser utilizadas por outros indivíduos fazendo apenas a tradução dos rótulos de cada entidade, um trabalho consideravelmente menos árduo que a recriação de uma ontologia de domínio.

4.2.7 OWL 2

Nesta seção abordamos a linguagem OWL e suas partes integrantes, como os *namespaces*, cabeçalhos, classes, propriedades, no entanto, o que foi apresentado se refere a versão de 2004, denominada apenas “OWL”. A documentação oficial, disponibilizada pelo W3C, já descreve uma versão mais atualizada, a “OWL 2”, que é uma versão revisada publicada em 2004.

A OWL 2 foi publicada em 2012 e é atualmente a última versão disponível, sendo que a estrutura da segunda versão é bastante similar a anterior. O núcleo RDF/XML da linguagem permanece inalterado, sendo assim a sintaxe de descrição de entidades e relacionamentos não sofreram mudanças.

O W3C destaca ainda que todas as ontologias construídas em OWL são totalmente válidas e compatíveis com a versão OWL 2, garantindo assim retrocompatibilidade entre as versões, algo de suma importância, pois a construção de ontologias é um processo custoso em vários sentidos.

Algumas das diferenças entre a primeira e a segunda versão são basicamente adição de funcionalidades que melhoram a expressividade da linguagem, como por exemplo:

- Cadeias de propriedades;
- Tipos de dados mais ricos;
- Restrições de cardinalidade qualificadas;
- Propriedades assimétricas e reflexivas;

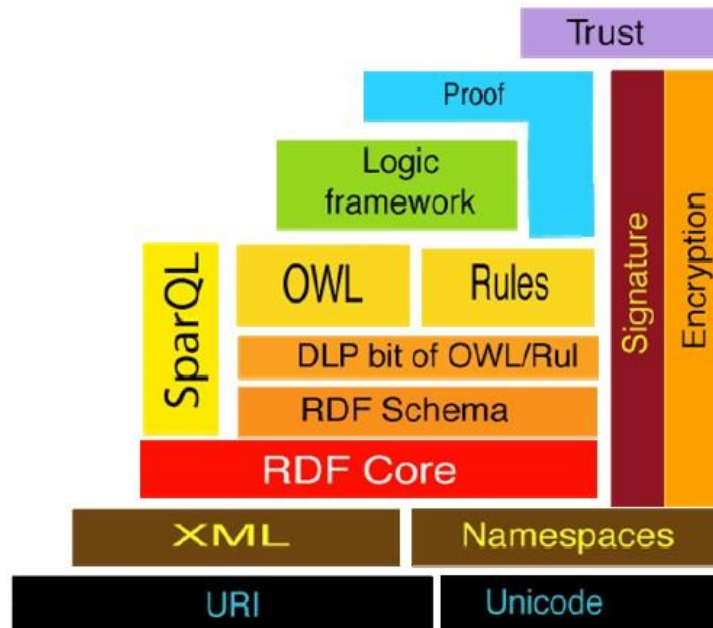
4.3 SPARQL

A linguagem de consulta SPARQL é definida Prud'Hommeaux (2008, tradução nossa) como uma linguagem para expressar consultas em diversas fontes de dados, desde que os dados estejam armazenados nativamente como RDF. Segundo Luz (2013) o termo SPARQL, cuja pronúncia é “sparkle” é um acrônimo recursivo para SPARQL Protocol and RDF Query Language corroborando com a definição de Prud'Hommeaux sobre a proposta de ser um mecanismo de consulta a dados estruturados no formato *Resource Description Framework* (RDF).

Filho e Lóscio (2009) colocam que a linguagem SPARQL é a linguagem recomendada pelo W3C para a recuperação de informações em documentos no formato RDF/RDFS, tal como a linguagem SQL é usada para recuperar registros em bases de dados estruturados.

Dentro do assunto da Web Semântica (WS) Ramalho (2006) apresenta uma sólida base histórica dos conceitos, propostas e tecnologias envolvidas no desenvolvimento da WS, detalhando por exemplo o uso do formato RDF e também do XML, isso na segunda proposta apresentada pelo World Wide Web Consortium (W3C) em meados de 2002. Já em 2005 em uma nova renovada proposta o W3C propõe uma nova arquitetura, que já engloba a linguagem SPARQL e também a OWL para criação de ontologias, detalhada na seção 4. Na Figura 17, Ramalho (2006) mostra como a proposta da WS contempla uma camada representação física dos dados ladeada pela SPARQL, demonstrando assim a recomendação do W3C para seu uso.

Figura 17 – Arquitetura da Web Semântica proposta em 2005



Fonte: adaptado de Ramalho (2006)

De acordo com Pérez, Arenas e Gutierrez (2006), uma consulta SPARQL consiste em três partes:

- *Pattern Matching*: inclui várias características interessantes de combinação de padrões de gráficos, a união de padrões, aninhamento, filtragem e a possibilidade de escolher a fonte dos dados.
- *Solution modifiers*: permite modificar os valores de saída padrão aplicando alguns operadores como PROJECTION, LIMIT, DISTINCT, ORDER e OFFSET.
- *Output*: podem ser de diferentes tipos, como: booleanas, seleções, gráficos e descrições de recursos.

É importante destacar que estas partes não são obrigatórias para o funcionamento de toda e qualquer consulta, sendo facultado seu uso dependendo das necessidades de cada situação. Na Figura 18 pode-se ver uma consulta simples, retirada dos exemplos de DuCharme (2013).

Figura 18 – Exemplo de consulta SPARQL

```
1
2 # filename: ex003.rq
3
4 PREFIX ab: <http://learningsparql.com/ns/addressbook#>
5
6 SELECT ?craigEmail
7 WHERE
8 { ab:craig ab:email ?craigEmail . }
9
```

Fonte: adaptado de DuCharme (2013)

A Figura 18 mostra uma consulta simples construída usando SPARQL, onde na segunda linha há um comentário, na quarta linha há a definição de um prefixo, um tipo de apelido que será usado mais abaixo no código e nas linhas seis a oito a consulta em si. Neste exemplo a instrução usada é SELECT indicando que o resultado será recuperado e apresentado em forma de uma seleção, o objeto de interesse é definido pela variável “?craigEmail” que aparece também na tripla da condição WHERE, neste caso indicando que o objetivo é recuperar o email de um recurso chamado “craig”. Este exemplo utiliza o formato *Turtle triple* que segue uma proposta mais simplista que os documentos no formato RDF/RDFS, no entanto, o uso de fontes de informação em RDF ou OWL pode ser feito apenas adicionando os respectivos prefixos e apontando a origem dos dados na execução da consulta.

Um processador SPARQL é um software capaz de rodar consultas em dados locais ou remotos DuCharme (2013, tradução nossa). Para os testes realizados foi utilizado o Jena framework, um programa baseado em Java capaz de realizar consultas em linguagem SPARQL em ontologias no formato OWL.

Para a proposta deste trabalho utilizaremos poucos recursos da linguagem SPARQL, visto que sua capacidade é muito maior que as necessidades apresentadas por esta pesquisa. A proposta é utilizar a instrução SELECT para retornar os resultados de uma busca por determinados relacionamentos contidos em uma ontologia no formato OWL. Detalhes e explicações dos códigos utilizados nas consultas serão abordados de forma mais cuidadosa no capítulo 6, junto com a proposta de indexação baseada em ontologia.

5

Método de expansão de termos de indexação baseado em ontologia

Neste capítulo descreve-se o conceito e o funcionamento do método de expansão de termos de indexação baseado em ontologias, apresentando na sequência os materiais utilizados no desenvolvimento do protótipo construído para simular o funcionamento do método. Os detalhes sobre os softwares, configurações e implementações são explicados e apresentados por meio de figuras para que outros autores sejam capazes de reproduzir a implementação do método em outros contextos, permitindo a incorporação da proposta em aplicações e/ou sistemas de indexação automática.

5.1 Conceito

Antes de abordar o conceito do método faz-se necessário resgatar algumas definições importantes, como o processo de indexação e as ferramentas que podem ser utilizadas em sua automação.

Gil Leiva (2008, p.63, tradução nossa) afirma que “o objetivo da indexação de documentos é permitir o armazenamento das informações para atender as necessidades informacionais dos indivíduos”, apresentando uma visão do processo de indexação composta basicamente por duas etapas:

1. Extração e mapeamento dos conceitos
2. Tradução

A definição do processo de indexação composta por duas etapas é descrita também por UNISIST (1981, p.85), Chaumier (1988, p.23), Fidel (1994, p.573) e Lancaster (2004, p.8-9),

demonstrando assim que esta é aceita por alguns autores, no entanto, há divergência entre pesquisadores, sendo que o processo chega a ter até cinco etapas segundo alguns.

As divergências na definição do processo de indexação residem principalmente na divisão entre as etapas indicadas por cada autor, sendo que essencialmente o processo é composto pelas etapas de extração e mapeamento de conceitos seguido de tradução, como coloca Gil Leiva (2008).

Quanto ao processo de indexação automática, observa-se que ele também segue estas etapas, diferindo na forma como os sistemas desempenham cada uma das etapas e quais ferramentas utilizam.

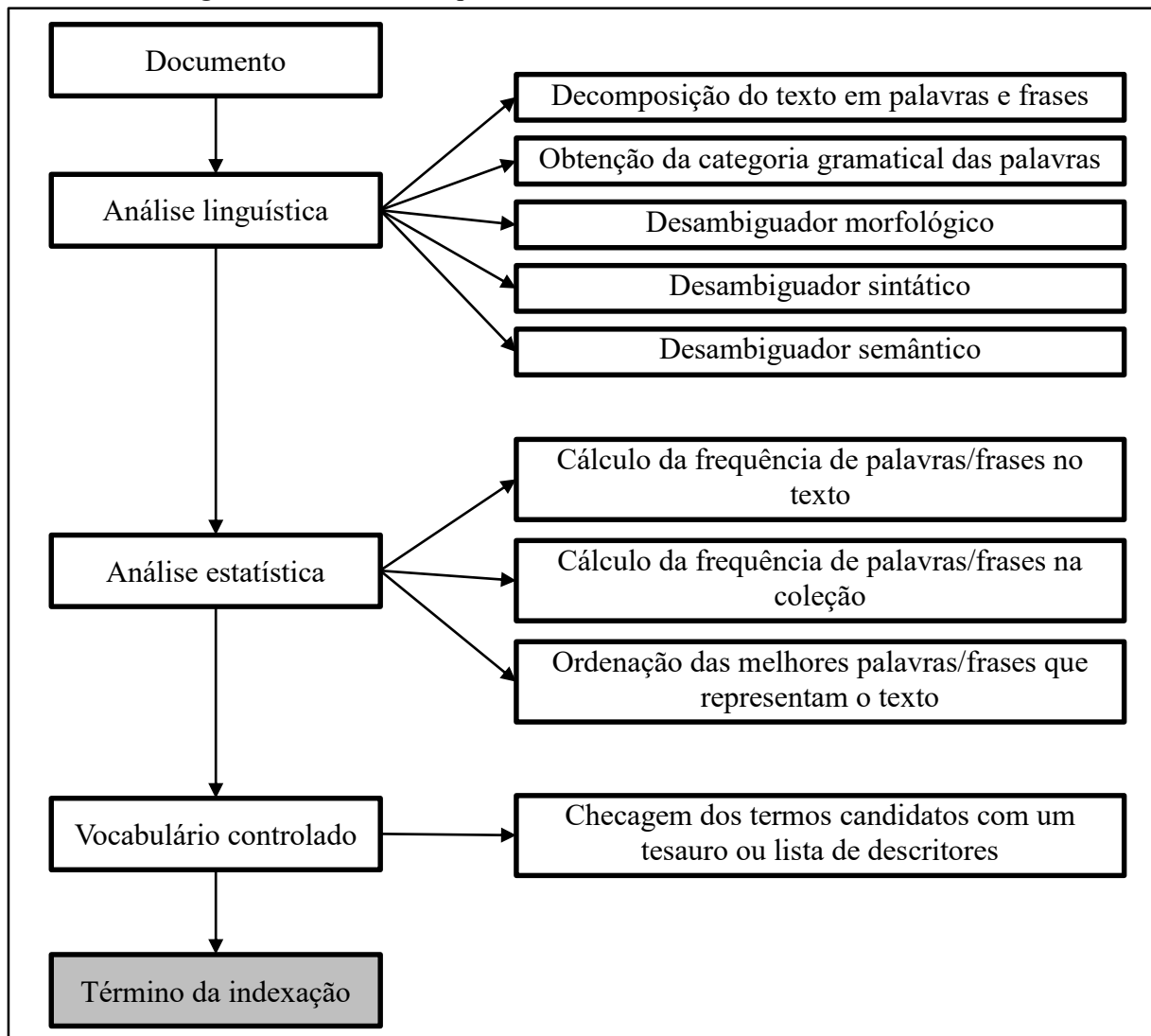
É possível classificar os sistemas de indexação automáticos como: estatísticos, linguísticos ou mistos, em função do valor agregado a frequência de palavras ou o tratamento da linguagem em sua concepção e desenvolvimento (GIL LEIVA, 2008, p.328).

Em relação ao processo de indexação automática, Gil Leiva (2008, p.368) afirma que não há um consenso, devido principalmente a complexidade do assunto, no entanto, o autor apresenta um modelo de arquitetura de sistema de indexação automática, que pode ser observado na Figura 19, onde algumas etapas comuns são apresentadas baseado em estudos de propostas levantadas pelo autor entre os anos 1950 e 2008.

A partir do modelo de arquitetura proposto por Gil Leiva é possível observar três momentos distintos, a análise linguística, responsável pelo tratamento primário das palavras ou frases do texto, a análise estatística, onde por meio de algoritmos e/ou regras pré-definidas são escolhidos os candidatos a descritores do documento e, por fim, a checagem destes utilizando um vocabulário controlado, etapa ao qual o autor cita o uso de tesouros ou listas de descritores.

Gil Leiva propõe ainda um sistema de indexação automática, denominado SISA, onde um tesouro é utilizado como ferramenta de controle terminológico na etapa de cotejamento dos termos candidatos. Narukawa (2011) faz uma análise comparativa dos resultados da utilização do SISA para indexação automática e da indexação manual de um corpo de documentos, concluindo que a utilização do tesouro como vocabulário controlado torna a solução susceptível de falhas, indicando ainda como uma possível solução o uso de ontologias em substituição ao tesouro.

Figura 19 – Modelo de arquitetura de um sistema de indexação automática



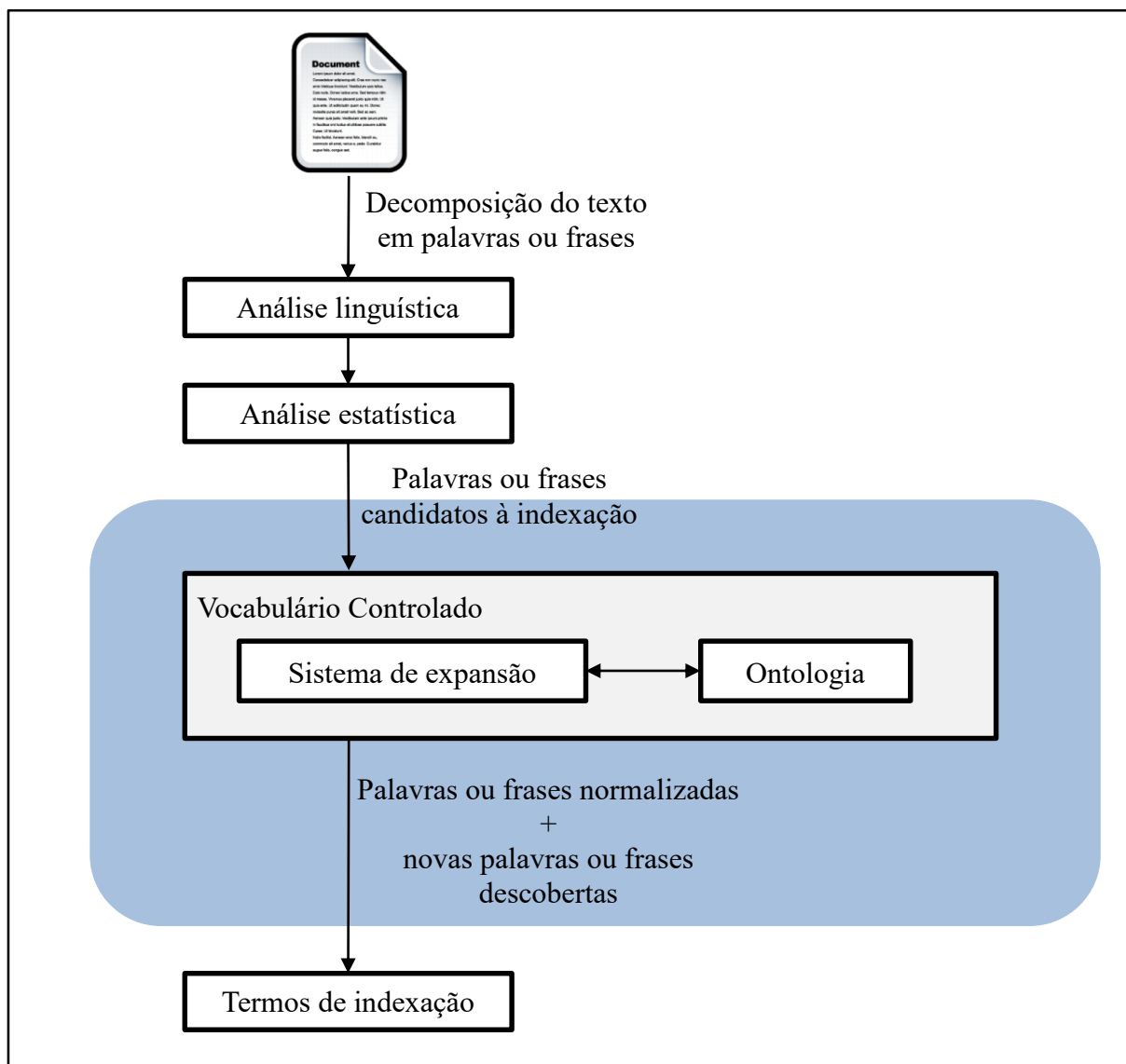
Fonte: adaptado de Gil Leiva (2008, p.367)

Desta forma o método proposto busca atacar essa problemática, pois, pode auxiliar no processo de indexação automática junto a etapa de cotejamento de termos no vocabulário controlado. Destaca-se que a proposta do trabalho não é desenvolver um sistema de indexação automática, mas um artefato auxiliar ao processo.

Considerando a Figura 19 o método proposto pode ser utilizado na terceira etapa, onde a arquitetura proposta por Gil Leiva apresenta uma etapa onde é feita uma checagem dos termos candidatos utilizando um tesauro ou uma lista de descritores, a proposta utiliza uma palavra ou frase como entrada e por meio do uso de uma ontologia de domínio resulta em uma ou mais palavras ou frases em sua saída.

Uma representação de um processo de indexação automática, baseado no modelo proposto por Gil Leiva (2008) é mostrada na Figura 20, sendo que o quadro em destaque representa uma possível aplicação do método proposto no processo de indexação automática.

Figura 20 – Modelo de arquitetura do processo de indexação automática utilizando o método proposto



Fonte: desenvolvido pelo autor

O funcionamento do método destacado no quadro da Figura 20 consiste em uma aplicação capaz de receber uma palavra ou frase, fazer seu processamento utilizando uma ontologia de domínio onde é feita uma expansão do candidato a descritor buscando novos descritores hierarquicamente genéricos, sinônimos e/ou equivalentes oferecendo também a possibilidade de fazer estes processamentos em todos os idiomas suportados pela ontologia. A aplicação resulta o termo original juntamente com os novos termos descobertos.

5.2 Desenvolvimento

A ontologia de domínio utilizada neste trabalho foi a PedTerm, uma ontologia que representa o domínio de conhecimento da pediatria apresentando informações relacionadas à saúde e ao desenvolvimento infantil desde o pré-natal até os 21 anos de idade.

Essa ontologia está disponível no BioPortal, um grande repositório de ontologias na área biomédica, que se declara como “o repositório mais abrangente do mundo de ontologias biomédicas”. Na Figura 21 pode-se ver a página inicial do repositório.

A principal razão da escolha desta ontologia é simular as propostas de Picker (2014), sendo que para isso foi necessária a adoção da mesma ontologia utilizada pela autora. No entanto, foi necessário obter o arquivo físico da ontologia (OWL) diretamente do seu repositório.

Figura 21 – Página inicial do Repositório de Ontologias BioPortal

The screenshot shows the BioPortal homepage. At the top, there's a navigation bar with links: BioPortal, Browse, Search, Mappings, Recommender, Annotator, Resource Index, Projects. On the right, there are links for 'Recently Viewed', 'Sign In', 'Help', and 'Feedback'. Below the navigation bar, a welcome message reads: 'Welcome to BioPortal, the world's most comprehensive repository of biomedical ontologies.' To the right of this message, it says 'Current Release: 4.25 (March 2016)'. Below the welcome message, there are three search boxes: 'Search all ontologies', 'Find an ontology', and 'Search resources'. Each search box has a text input field and a 'Search' button. Below the search boxes, there are three main sections: 'Ontology Visits (February 2016)', 'Latest Notes', and 'Latest Mappings'. The 'Ontology Visits' section lists several ontologies with their visit counts: Current Procedural Terminology (CPT) - 35165, RxNORM (RXNORM) - 19288, Medical Dictionary for Regulatory Activities (MEDDRA) - 15510, Systematized Nomenclature of Medicine - Clinical Terms (SNOMEDCT) - 10927, and National Drug Data File (NDDF) - 6307. The 'Latest Notes' section contains two notes about the 'centiMorgan' unit and 'Grade definition (Common Terminology Criteria for Adverse Events)'. The 'Latest Mappings' section lists several mappings between ontologies, such as 'Health Entities (HEIO) <=> Site of care (SNOMEDCT)' and 'Site of care (SNOMEDCT) <=> Health Entities (HEIO)'. At the bottom of the page, there is a footer with the text: 'The National Center for Biomedical Ontology is one of the National Centers for Biomedical Computing supported by the NHGRI, the NHLBI, and the NIH Common Fund under grant U54-HG004028. Copyright © 2005-2016, The Board of Trustees of Leland Stanford Junior University. All rights reserved.' and links to 'NCBO Website', 'Release Notes', 'Terms of Use', 'Privacy Policy', and 'How to Cite'.

Fonte: Repositório de Ontologias BioPortal⁵

Após localizar a ontologia PedTerm – *Pediatric Terminology*, é apresentada uma página com os detalhes sobre esta ontologia, como o formato da ontologia, neste caso OWL, a categoria, “Saúde Humana” além de uma breve descrição:

⁵ Disponível em: <http://bioportal.bioontology.org/> Acesso em: 06/04/2016

Termos relacionados com a pediatria, o que representa informações relacionadas com a saúde e desenvolvimento da criança desde a pré-natal até os 21 anos de idade; Contribuições do Instituto Nacional de Saúde Infantil e Desenvolvimento Humano. (Whetzel *et al*, 2011, tradução nossa).

A ontologia é composta por 1771 classes e teve seu desenvolvimento iniciado em 2011, sendo que uma versão 2.0 foi disponibilizada em 2013, sendo esta a versão utilizada neste trabalho. Na Figura 22 pode-se observar o início do arquivo OWL, onde é declarado o documento RDF como todos os *namespaces* necessários à ontologia e na sequência a *tag* OWL que dá início à ontologia.

Figura 22 – Cabeçalho da ontologia PedTerm

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://www.owl-ontologies.com/Ontology1358660052.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
  xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://www.owl-ontologies.com/Ontology1358660052.owl">
  <owl:Ontology rdf:about=""/>

  <owl:Class rdf:ID="Disease_or_Disorder">
    <ID rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >C2991</ID>
    <NCI_PT rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Disease or Disorder</NCI_PT>
    <NCI_Definition rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Any abnormal condition of the body or mind that causes discomfort, dysfunction, or dis
  <SYN rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Disease</SYN>
    <Subset_Association1 rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >NICHD Pediatric Terminology</Subset_Association1>
    <Subset_Association2 rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Neonatal Research Network Terminology</Subset_Association2>
    <rdfs:label xml:lang="en">Disease or Disorder</rdfs:label>
    <rdfs:label xml:lang="es">Emfermedad o Transtorno</rdfs:label>
    <rdfs:label xml:lang="pt">Doença ou Distúrbio</rdfs:label>
  </owl:Class>
```

Fonte: adaptador da ontologia PedTerm

A PedTerm foi criada originalmente no idioma Inglês com suas classes e propriedades nomeadas apenas usando o atributo ID, não foram utilizados rótulos (propriedade *label*) OWL para descrever suas entidades. Isto configura um problema para a abordagem proposta, pois os métodos apresentados utilizam os rótulos para comparar os termos do documento com o vocabulário controlado.

Desta forma foi necessário fazer alguns ajustes na ontologia original, inserindo os *labels* em algumas classes. Nesta etapa houve-se o cuidado de criar rótulos na língua original da

ontologia (Inglês) e também em português e espanhol, para que fosse possível testar as propostas de indexação multilíngue.

Na Figura 23 pode-se ver um trecho da ontologia PedTerm com a classe “Disease_or_Disorder”, nas linhas 35 a 43 podem ser vistos os rótulos adicionados na classe. Em cada *tag* “*rdfs:label*” há uma propriedade “*xml:lang*” com a declaração do idioma deste rótulo. Podem ser informados quantos rótulos sejam necessários para cada situação.

Figura 23 – Trecho da ontologia PedTerm, exemplo de criação dos rótulos (*labels*)

```
15 <owl:Class rdf:ID="Disease_or_Disorder">
16   <ID rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
17     C2991
18   </ID>
19   <NCI_PT rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
20     Disease or Disorder
21   </NCI_PT>
22   <NCI_Definition rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
23     Any abnormal condition of the body or mind that causes discomfort, dysfunc
24   </NCI_Definition>
25   <SYN rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
26     Disease
27   </SYN>
28   <Subset_Association1 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
29     NICHD Pediatric Terminology
30   </Subset_Association1>
31   <Subset_Association2 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
32     Neonatal Research Network Terminology
33   </Subset_Association2>
34   <!-- EP -->
35   <rdfs:label xml:lang="en">
36     Disease or Disorder
37   </rdfs:label>
38   <rdfs:label xml:lang="es">
39     Enfermedad o Trastorno
40   </rdfs:label>
41   <rdfs:label xml:lang="pt">
42     Doença ou Distúrbio
43   </rdfs:label>
44   <!-- EP -->
45 </owl:Class>
```

Fonte: Desenvolvido pelo autor com base na ontologia PedTerm

A declaração dos rótulos foi feita apenas nas classes necessárias à execução dos exemplos apresentados no Capítulo 6, sendo que alterar todas as classes pertencentes a esta ontologia não era uma tarefa pertinente aos propósitos da pesquisa.

Após a obtenção e configuração da ontologia foi providenciado um mecanismo capaz de permitir a execução de consultas em linguagem SPARQL.

Segundo Ducharme (2013, p.5) uma grande variedade de processadores SPARQL estão disponíveis para a execução de consultas em dados armazenados local ou remotamente. Segundo o autor os termos “*SPARQL Processor*” ou “*SPARQL Engine*” são utilizados

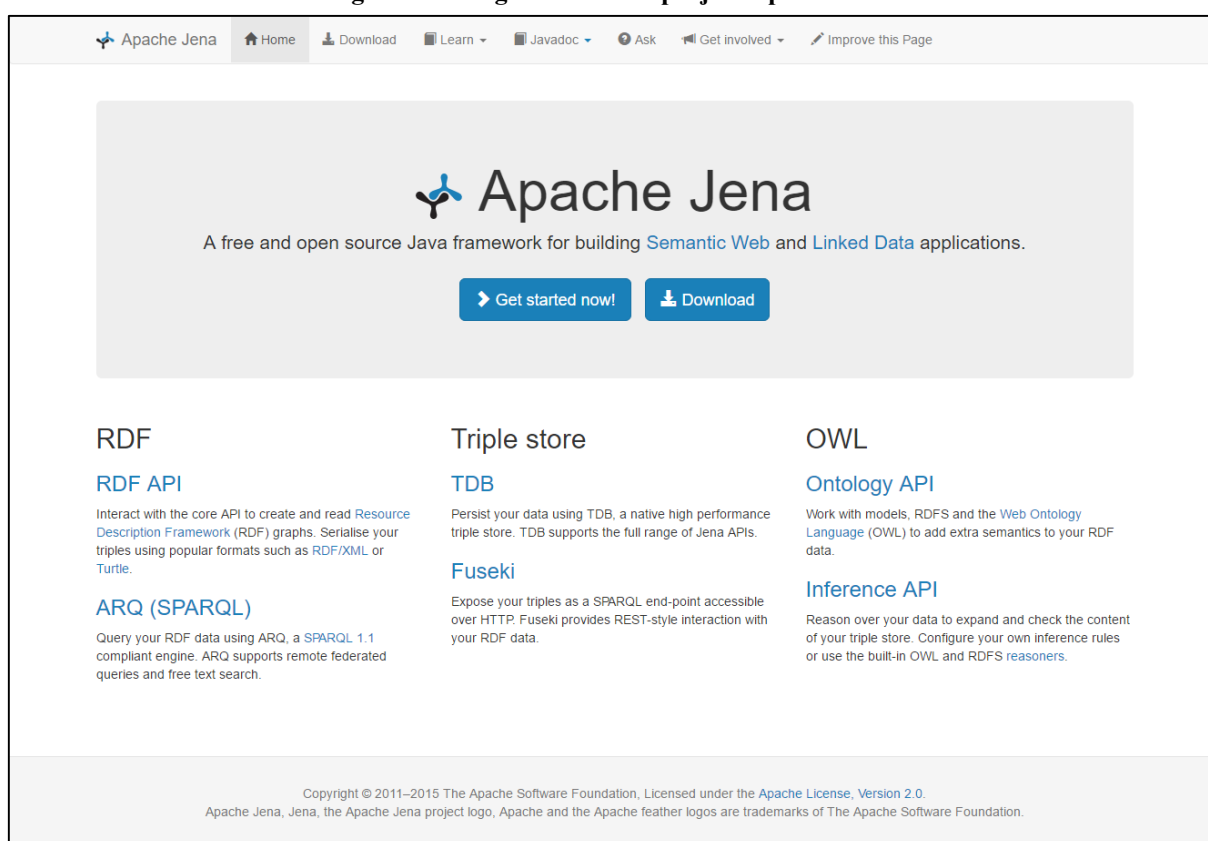
indistintamente para se referir a uma mesma coisa, um programa que é capaz de aplicar uma consulta SPARQL em uma base de dados e apresentar o resultado.

Ducharme (2013) ainda apresenta um programa em Java, o processador ARQ, que é utilizado para a execução dos exemplos apresentados em seu livro “*Learning SPARQL: Querying and Updating with SPARQL 1.1*”, assim, seguindo as recomendações do autor, processador ARQ foi utilizado nesta pesquisa.

O processador ARQ é parte integrante do Apache Jena⁶, que por sua vez pode ser definido como um “*framework* livre e de código aberto, feito em Java é voltado à construção de aplicações para web semântica e *linked data*” Apache Software Foundation (2015).

Na Figura 24 pode-se observar a página inicial do Apache Jena, nela há *links* para o processador ARQ e também para o Fuseki, que também é utilizado neste trabalho.

Figura 24 – Página inicial do projeto Apache Jena



Fonte: Projeto Apache Jena⁷

⁶ <https://jena.apache.org/index.html>

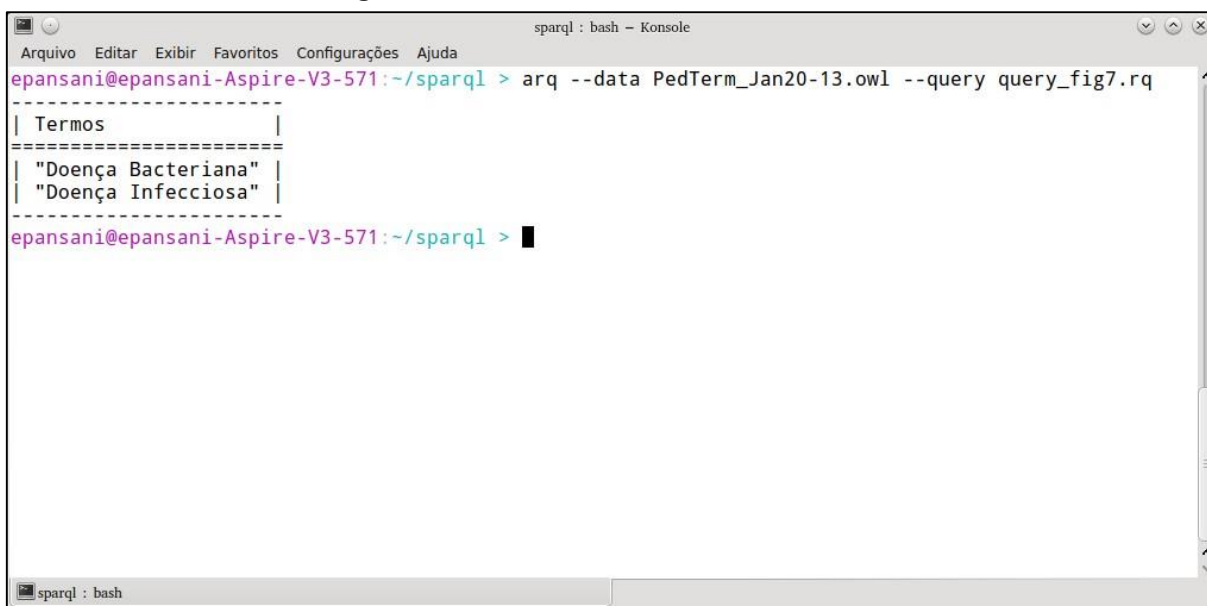
⁷ Disponível em: <https://jena.apache.org/> Acesso em: 06/04/2016

O processador ARQ é responsável por permitir a execução das consultas construídas em linguagem SPARQL. Por esta razão ele foi descarregado e configurado como um serviço no sistema operacional do computador onde foram realizados os testes.

Uma etapa que precedeu o desenvolvimento do protótipo foi a execução de algumas consultas SPARQL visando compreender a sintaxe da linguagem e testar as configurações feitas na ontologia, desta forma foram executadas algumas instruções no terminal de comandos do sistema operacional Linux, ambiente onde o protótipo foi implementado.

Na Figura 25 pode-se observar o resultado da execução da consulta SPARQL da Figura 32, que é detalhada no Capítulo 6. Na Figura 25 é possível ver que há uma instrução “arq” com dois parâmetros “--data” que indica qual fonte de dados será utilizada para a busca e o segundo “--query” indica o arquivo onde está salva a consulta em linguagem SPARQL. O resultado da execução é apresentado nas linhas seguintes a execução da instrução.

Figura 25 – Execução de uma consulta SPARQL

A terminal window titled 'sparql : bash - Konsole' showing a command prompt. The user 'epansani@epansani-Aspire-V3-571' is in the directory '~/sparql'. They execute the command 'arq --data PedTerm_Jan20-13.owl --query query_fig7.rq'. The output is a table with two rows: 'Doença Bacteriana' and 'Doença Infecciosa'.

```
sparql : bash - Konsole
epansani@epansani-Aspire-V3-571:~/sparql > arq --data PedTerm_Jan20-13.owl --query query_fig7.rq
-----
| Termos          |
-----
| "Doença Bacteriana" |
| "Doença Infecciosa" |
-----
epansani@epansani-Aspire-V3-571:~/sparql > █
```

Fonte: Terminal de comandos do Sistema Operacional Linux Mint 17

Ao analisar a Figura 25 pode-se constatar que é possível programar um sistema para realizar inferências automáticas utilizando uma ontologia OWL como um vocabulário controlado. No entanto, a forma com que a consulta é executada é bastante complexa, sendo necessária a criação de uma interface mais amigável. Desta forma, a próxima etapa foi a instalação do Fuseki, um servidor SPARQL que serve de base para a construção de uma interface mais usável.

Para a construção do protótipo utilizado nesta pesquisa, foi escolhida uma abordagem de uma aplicação web, ou seja, um software que é acessado utilizando um navegador por meio de um endereço do tipo “www”. As razões para a escolha desta tecnologia são:

- Independência de sistema operacional, é possível utilizar um navegador em computadores com diferentes sistemas operacionais e também em dispositivos móveis como *smartphones* e *tablets*.
- Não necessitam de instalação, o programa não necessita de ser instalado e configurado no computador do usuário, facilitando o uso e também a atualização em caso de necessidade.
- Intimidade do autor, esta modalidade de programação era previamente conhecida, logo houve mais facilidade ao desenvolver o sistema.
- Possibilidade de criação de um *webservice*, esta é a maior motivação da escolha desta abordagem, pois uma aplicação web pode se transformar em um *webservice*, permitindo assim sua integração com outros aplicativos e/ou sistemas.

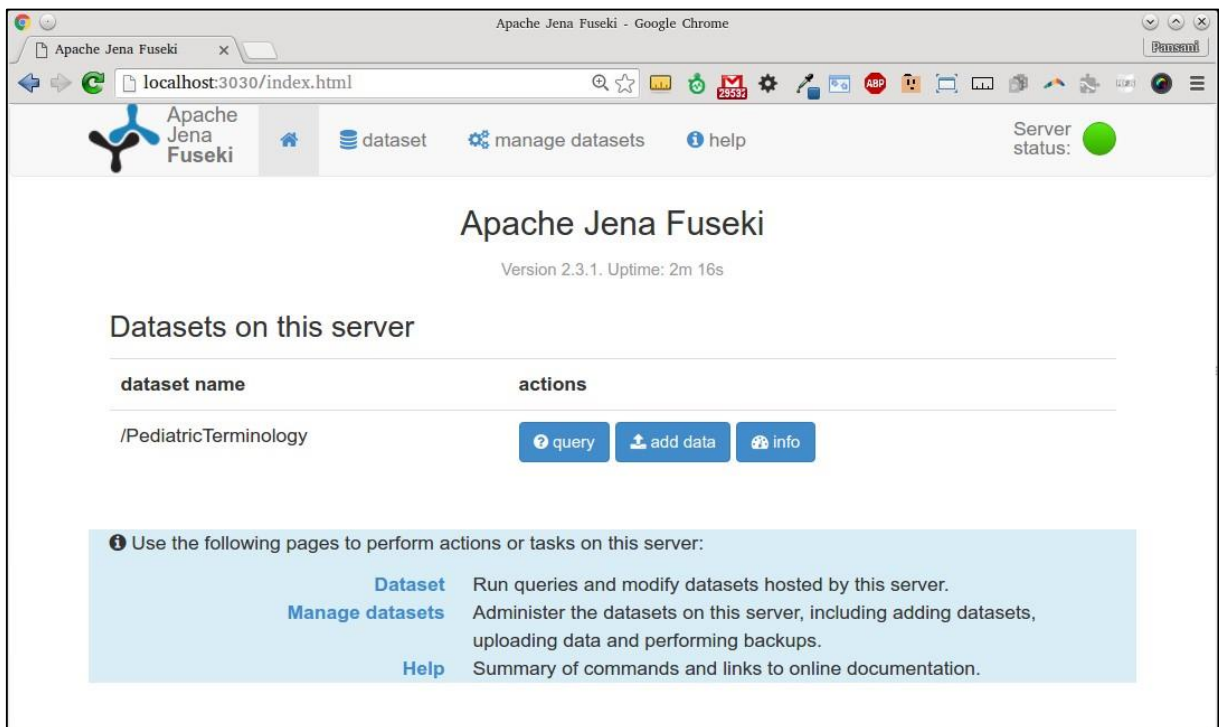
A proposta do protótipo desenvolvido vai além de validar as ideias de Pickler (2014) e colaborar com os estudos em indexação automática, vislumbrando ser parte integrante do projeto OntoSmart, proposto por Ferneda (2013), justificando assim a escolha da tecnologia que pode oferecer integração entre sistemas distintos.

Após descarregar⁸ e configurar⁹ o pacote Fuseki pode-se acessar o servidor pelo endereço local na porta 3030, como é possível ver no endereço do navegador mostrado na Figura 26. Essa é a tela inicial do servidor Fuseki, já há um *dataset* cadastrado “*PediatricTerminology*”, pois esta é uma captura do servidor já configurado para a aplicação desenvolvida.

⁸ <https://jena.apache.org/download/index.cgi>

⁹ <https://jena.apache.org/documentation/fuseki2/fuseki-configuration.html>

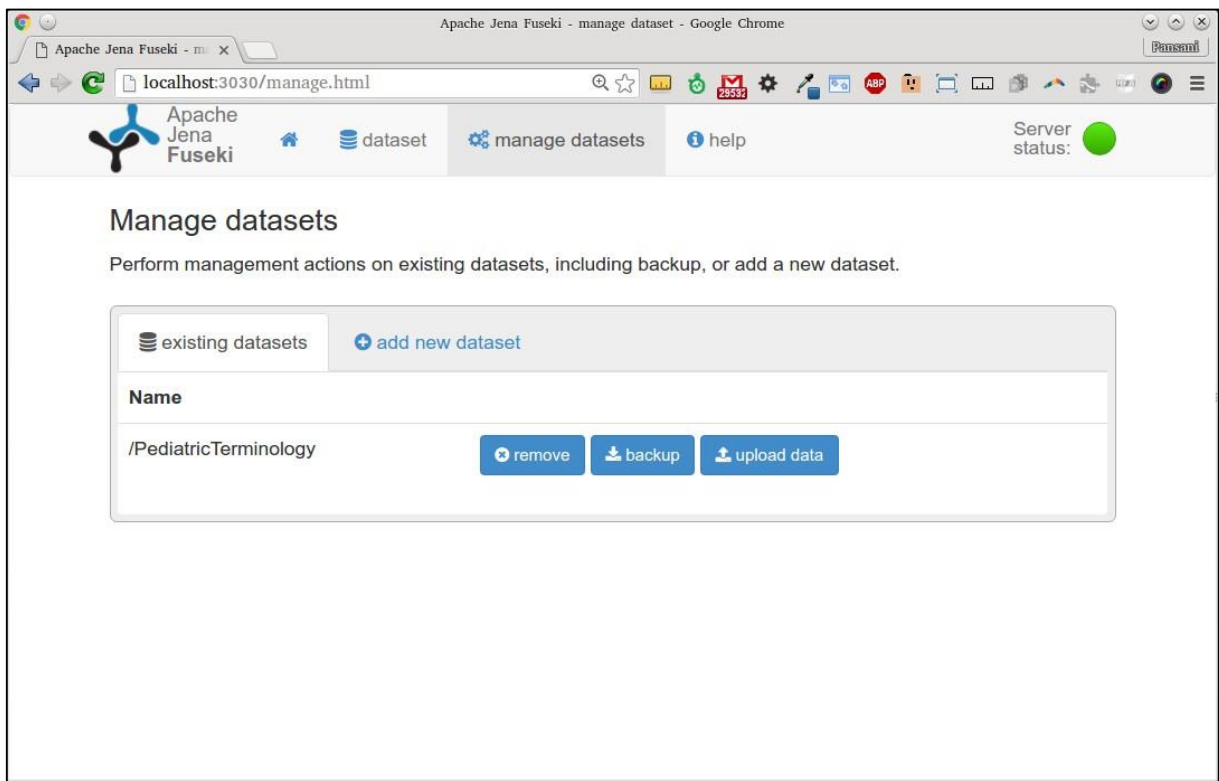
Figura 26 – Página inicial do servidor Apache Jena Fuseki



Fonte: Captura de tela da aplicação Apache Jena Fuseki

Em uma nova instalação nenhum *dataset* será apresentado, sendo necessário ir para a área de administração de bases de dados (*manage datasets*), onde é possível ver as bases existentes e cadastrar novas caso seja necessário. Para se criar uma nova base é necessário informar apenas um nome para a base e o tipo de persistência, onde há duas opções “em memória” um tipo de base de dados que é perdida ao desligar o servidor e outra “persistente” o tipo utilizado neste trabalho, onde a base de dados é criada e armazenada de forma permanente.

Figura 27 – Página de administração das bases de dados do servidor Apache Jena Fuseki



Fonte: Captura de tela da aplicação Apache Jena Fuseki

A Figura 27 mostra a página de administração de bases de dados do servidor Fuseki, é possível ver duas abas, a primeira listando as bases existentes e suas respectivas opções e a segunda mostra a opção de criar uma nova base de dados. Após criar uma base é preciso preenche-la com dados, para isso utiliza-se a opção “upload data” que permite escolher um ou mais arquivos no formato XML padrão OWL, RDF ou RDF *turtle*. Após enviar um ou mais arquivos o servidor cria uma estrutura interna que permite efetuar as consultas a base de dados utilizando as relações contidas nos arquivos enviados.

A base de dados “*PediatricTerminology*” foi preenchida com a ontologia PedTerm já com os rótulos criados, desta forma todo o conhecimento contido na ontologia ficou disponível para consultas no Fuseki.

Após preparar a base de dados é possível efetuar consultas SPARQL via interface do servidor Fuseki, como pode ser visto na Figura 28, onde há algumas opções para auxiliar na criação da consulta, como a adição de prefixos e a seleção de exemplos de consultas como “selecionar triplas” ou “selecionar classes”.

Figura 28 – Interface de consultas a base *PediatricTerminology*

Apache Jena Fuseki

dataset manage datasets help

Server status: ●

Dataset: /PediatricTerminology

query upload files edit info

SPARQL query

To try out some SPARQL queries against the selected dataset, enter your query here.

EXAMPLE QUERIES

Selection of triples Selection of classes

PREFIXES

rdf rdfs owl xsd

SPARQL ENDPOINT: http://localhost:3030/PediatricTerminology/

CONTENT TYPE (SELECT): XML

CONTENT TYPE (GRAPH): Turtle

```
1 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2
3 SELECT (str(?label) as ?Termos)
4
5 WHERE
6 {
7   {
8     ?res1 rdfs:label "Tétano"@pt .
9     ?res1 rdfs:subClassOf ?res2 .
10    ?res2 rdfs:label ?label
11    FILTER(lang(?label) = "pt") .
12   }
13   UNION
14   {
```

QUERY RESULTS

Raw Response Table

Search:

Show 50 entries

	Termos
1	"Doença Bacteriana"
2	"Doença Infecciosa"

Showing 1 to 2 of 2 entries

Fonte: Captura de tela da aplicação Apache Jena Fuseki com uma consulta elaborada pelo autor

A página apresentada na Figura 28 pode ser acessada selecionando a ação “query” que pode ser vista na Figura 26, logo à frente no nome do *dataset*. Essa interface de consulta possui um campo de texto que está localizado no centro da Figura 28, nele o usuário pode testar

diferentes consultas SPARQL e executá-las diretamente no servidor, sem a necessidade de desenvolvimento de algum software ou aplicação.

Para demonstrar o funcionamento deste recurso utilizamos novamente a consulta SPARQL da Figura 32, que pode ser vista no Capítulo 6, o resultado da execução da consulta é apresentado logo abaixo do campo texto (painel denominado “*query results*”), onde é possível ver que o resultado é idêntico ao apresentado na Figura 25.

É importante esclarecer que a interface do servidor Fuseki é voltada a administração das bases de dados para aplicações de Web Semântica, sendo que em um sistema funcionando em ambiente de produção este acesso às ferramentas de administração e consulta disponibilizados pelo servidor devem ter seu acesso restrito por meio da configuração do Fuseki. Estas configurações e restrições de segurança não foram implementadas no protótipo desenvolvido para esta pesquisa devido à aplicação ainda não ter sido disponibilizada publicamente na web.

A instalação e configuração dos softwares ARQ e Fuseki pertencentes ao Apache Jena pode ser compreendida como a implementação de requisitos técnicos para o desenvolvimento do protótipo de expansão de termos. Agora são abordadas de forma sucinta as linguagens de programação utilizadas para criar o protótipo apresentado no Capítulo 6.

Para a criação da aplicação Web foi utilizada a linguagem de programação PHP na versão 5.5.9 e a linguagem de marcação HTML na versão 5. A linguagem PHP é um módulo que precisa de um servidor para ser acoplada, para este propósito foi utilizado o Apache HTTP Server¹⁰.

O servidor Apache HTTP é um servidor Web bastante utilizado para aplicações web tanto em ambientes comerciais quanto educacionais, é um software livre e tem mecanismos que facilitam bastante sua instalação em sistemas operacionais Linux, como foi utilizado neste trabalho.

PHP é um acrônimo para “*PHP: Hypertext Preprocessor*”, é uma linguagem de programação de ampla utilização, interpretada, especialmente interessante para o desenvolvimento para a web. É comumente mesclada com o HTML com objetivo de tornar as páginas dinâmicas, alterando seu conteúdo antes de enviar as páginas ao navegador.

¹⁰ <https://httpd.apache.org/>

A linguagem PHP é uma linguagem *server side*, sendo algumas de suas funções permitir o envio e recebimento de dados via formulários além de permitir a conexão e a manipulação de informações provenientes de um banco de dados qualquer.

Já a linguagem de marcação HTML tem por objetivo estruturar e apresentar as informações e a interface do sistema, sendo que utilizar HTML muitas vezes implica em utilizar também as linguagens CSS e JavaScript, responsáveis respectivamente pelo visual e controle de eventos e ações nas páginas HTML.

Para agilizar o desenvolvimento foi utilizado o *Framework Bootstrap*¹¹ que é um framework que reúne HTML, CSS e JavaScript facilitando o desenvolvimento de projetos responsivos para Web. Além disso foi utilizada a biblioteca JavaScript jQuery¹² para facilitar a programação na respectiva linguagem.

A página inicial do protótipo desenvolvido por ser vista na Figura 29. Para o completo funcionamento uma biblioteca chamada SparqlLib.php¹³ foi utilizada para permitir que a linguagem PHP se conectasse ao servidor Fuseki, executando assim as consultas SPARQL geradas pelo formulário de entrada de dados da aplicação.

O funcionamento do sistema segue o seguinte fluxo:

1. Escolher o domínio;
2. Informar o termo a ser buscado no Vocabulário Controlado
 - a. Informar o idioma do termo
3. Escolher quantos níveis de exaustividade o sistema deverá buscar
4. Marcar o tipo de busca a ser feita
5. Clicar no botão “Buscar” para iniciar o processamento

Após iniciar o processamento, internamente o programa gera uma consulta SPARQL, sendo que os parâmetros informados no formulário de entrada formatam essa consulta, sendo, o domínio escolhido a base de dados ao qual a busca será feita.

¹¹ <http://getbootstrap.com/>

¹² <https://jquery.com/>

¹³ <http://graphite.ecs.soton.ac.uk/sparqlLib/>

Depois de gerar dinamicamente a consulta, esta é submetida ao servidor Fuseki, que por sua vez busca em sua base de dados por relações de acordo com o tipo de busca informado. Lembrando que a base de dados foi originada pela ontologia de domínio, que neste trabalho funciona como um vocabulário controlado.

Figura 29 – Página inicial do protótipo desenvolvido

OntoSmart

Início Sobre Contatos

Indexação Automática baseada em Ontologias

Início / Indexação Automática baseada em Ontologias

Domínio [Selecione o domínio para começar]

Termo Informe um dos termos extraídos do documento

Idioma do termo [Selecione o idioma do termo informado] Níveis de exaustividade [Selecione nível]

Tipo de busca ☒ Termos Genéricos ☒ Sinônimos ou Equivalentes ☐ Cross Language

Buscar Limpar

OntoSmart
Ontology-based Information Retrieval System

OntoSmart é um sistema de recuperação de informação Web, um mecanismo de busca (search engine) que utiliza a estrutura formal do Modelo Espaço Vetorial associado ao uso de ontologias como ferramenta de normalização terminológica da representação tanto dos documentos como das buscas.

unesp
UNIVERSIDADE ESTADUAL PAULISTA
"JULIO DE MESQUITA FILHO"

Fonte: desenvolvido pelo autor

No Capítulo 6 são apresentados exemplos de uso da aplicação desenvolvida, onde são propostas situações hipotéticas, as consultas geradas dinamicamente, os resultados da execução e uma breve discussão destes resultados em relação aos processos de indexação automática.

6

A utilização do protótipo na expansão de termos

No capítulo 3 foram abordados os principais conceitos de alguns métodos de indexação e também os tipos de indexação: por extração automática e por atribuição automática.

O método proposto por este trabalho utiliza ontologias como ferramentas de controle terminológico em uma indexação por atribuição. No entanto, antes de efetuar a atribuição de termos a um determinado documento é necessária uma etapa de extração, sendo que primeiramente são extraídos do documento os candidatos a descritores, para depois submeter estes ao sistema que é responsável por indicar outros termos relacionados ao descritor candidato.

De forma geral a extração automática de termos pode ser realizada por métodos puramente matemáticos (estatísticos) ou pela utilização de técnicas de Processamento de Linguagem Natural, porém o estudo destas tarefas e sua implementação fogem ao escopo desta pesquisa.

A utilização de ontologias no processo de indexação automática implica em determinar o assunto dos documentos (*corpus*), sendo assim, ao definir que um documento está relacionado a uma ontologia de domínio define-se implicitamente que o documento trata do assunto desta ontologia.

Porém é importante observar que um documento pode tratar sobre diversos assuntos, sendo muitas vezes necessário que mais de uma ontologia seja usada em sua indexação. Pickler (2014) afirma que tal problema pode ser solucionado com a possibilidade de um sistema atribuir a um mesmo documento ou *corpus* mais de uma ontologia para sua indexação, ou, ainda,

realizar uma junção de duas ou mais ontologias utilizando técnicas de interoperabilidade pesquisadas no âmbito da Ciência da Computação, como combinação, alinhamento, integração e mapeamento.

Para a proposta deste estudo assume-se que os termos em questão e a ontologia são de um mesmo domínio, sendo que nenhuma abordagem de descoberta automática de assunto ou utilização de múltiplas ontologias é explorada nessa pesquisa.

Neste capítulo demonstra-se a utilização do protótipo para expandir termos hipoteticamente extraídos de documentos textuais. Primeiramente são explicadas as propostas e resultados esperados da execução de cada teste, em seguida é apresentada uma figura com a interface do sistema e os parâmetros de entrada necessários a execução daquela proposta e por fim é apresentada a consulta em linguagem SPARQL que foi gerada internamente no sistema, para que seja possível compreender melhor cada proposta.

O funcionamento do protótipo implementado consiste em primeiramente escolher a ontologia de domínio por meio do campo “Domínio”, o primeiro campo que pode ser visto na Figura 29. Depois deve-se informar no campo “Termo” uma palavra que seria um candidato a descritor hipotético, em seguida escolhe-se o idioma do termo informado, para que o sistema saiba em quais rótulos fazer a busca, e por fim deve-se selecionar o nível de exaustividade.

O nível de exaustividade se refere a quantos níveis hierárquicos o sistema deverá buscar, caso seja escolhido um nível, apenas uma classe hierarquicamente superior será analisada, caso seja escolhido dois níveis, duas classes hierarquicamente superiores serão analisadas, e assim sucessivamente.

A última etapa é escolher no campo “tipo de busca” qual o tipo de expansão de termos que deve ser executada pelo sistema. Ao marcar a opção “Termos Genéricos” o sistema irá buscar por termos hierarquicamente mais genéricos do que o termo informado, obedecendo ao nível de exaustividade previamente escolhido. A opção “Sinônimos ou Equivalentes” procura em classes equivalentes e em classes com dois ou mais rótulos que representam um mesmo conceito, resultando assim em sinônimos ou em termos equivalentes em relação ao informado.

A última opção “*Cross Language*” faz com que o resultado não seja apresentado apenas no idioma informado, ao invés de filtrar os resultados para o idioma informado todos os rótulos de todas as linguagens contempladas pela ontologia são apresentados.

Nas subseções seguintes demonstra-se com maior riqueza de detalhes a atribuição automática de conceitos, a descoberta de termos sinônimos e também a indexação multilíngue.

6.1 Atribuição de conceitos

A atribuição de conceitos baseada em ontologia funciona da seguinte forma: cada termo extraído do texto deve ser submetido por meio de uma consulta SPARQL a uma ontologia, buscando coincidência entre este termo e o rótulo de alguma classe. Na ocorrência de tal coincidência, deve-se considerar o ID da classe à qual a propriedade label está associada para, a partir daí, realizar inferências ou traçar relacionamentos com outras classes da ontologia.

A ideia de se traçar relacionamentos por meio da classe encontrada funciona utilizando a estrutura taxonômica *rdfs:subClassOf* da linguagem OWL abordada no capítulo 5. Quando há uma coincidência entre um termo e uma classe pega-se esta classe e por meio da propriedade *subClassOf* verifica-se se ela possui uma superclasse (uma classe mais genérica), em caso positivo o rótulo desta nova classe é selecionado para ser atribuído a este documento, repetindo este processo quantas vezes for interessante.

No protótipo é possível escolher entre ou dois níveis de extensividade para se buscar superclasses e novos conceitos a serem atribuídos. O valor limite de dois foi fixado visando padronizar os testes, mas não há indícios de limitações técnicas para uma quantidade de níveis maior.

Na Figura 30 pode-se observar na área superior esquerda a representação de um documento hipotético de onde foi extraído por algum método de extração automática o termo “Tétano”, mostrado abaixo da figura que representa o documento. Ainda é possível observar uma pequena dobra no canto superior direito da página do documento a sigla “pt” que informa que este texto hipotético é do idioma português. Nesta mesma figura pode ser observado um trecho da ontologia PedTerm, este código é a linguagem OWL que declara as classes “*Disease_or_Disorder*”, “*Infectious_Disease*”, “*Bacterial_Disease*” e “*Tetanus*” juntamente com suas respectivas propriedades.

No entanto o propósito principal da Figura 30 é demonstrar por meio das elipses, retângulos e setas como é feito o processamento interno do sistema quando um termo é informado como entrada, neste caso o termo “Tétano”. A partir da comparação do termo com os rótulos das classes pertencentes à ontologia busca-se uma coincidência, quando encontrando

um rótulo idêntico ao termo o sistema infere que este termo pertence ao vocabulário e passa para a próxima etapa, que neste caso é buscar outras classes hierarquicamente mais genéricas.

O processamento exemplificado encontrou o rótulo “Tétano”, mostrado na figura com um retângulo ao redor da palavra “Tétano” na classe com a identificação (ID) “*Tetanus*”, então o sistema procura nessa classe se há uma declaração de hierarquia, no exemplo há, uma das propriedades da classe é *subClassOf* que indica que esta classe é uma subclasse da classe “*Bacterial_Disease*”, fato que pode ser observado na figura por meio de uma elipse ao redor da identificação “*#Bacterial_Disease*”.

Figura 30 – Exemplo de Indexação automática a partir de um termo de indexação



Tétano
Doença bacteriana
Doença Infecciosa

```
<owl:Class rdf:ID="Disease_or_Disorder">
  <rdfs:label xml:lang="en">Disease or Disorder</rdfs:label>
  <rdfs:label xml:lang="es">Enfermedad o trastorno</rdfs:label>
  <rdfs:label xml:lang="pt">Doença ou distúrbio</rdfs:label>
</owl:Class>

<owl:Class rdf:ID="Infectious_Disease">
  <rdfs:label xml:lang="en">Infectious Disease</rdfs:label>
  <rdfs:label xml:lang="es">Enfermedad Infecciosa</rdfs:label>
  <rdfs:label xml:lang="pt">Doença Infecciosa</rdfs:label>

  <rdfs:subClassOf rdf:resource="#Disease_or_Disorder"/>
</owl:Class>

<owl:Class rdf:ID="Bacterial_Disease">
  <rdfs:label xml:lang="en">Bacterial Disease</rdfs:label>
  <rdfs:label xml:lang="es">Enfermedad Bacteriana</rdfs:label>
  <rdfs:label xml:lang="pt">Doença bacteriana</rdfs:label>

  <rdfs:subClassOf rdf:resource="#Infectious_Disease"/>
</owl:Class>

<owl:Class rdf:ID="Tetanus">
  <rdfs:label xml:lang="en">Tetanus</rdfs:label>
  <rdfs:label xml:lang="es">Tétanos</rdfs:label>
  <rdfs:label xml:lang="pt">Tétano</rdfs:label>

  <rdfs:subClassOf rdf:resource="#Bacterial_Disease"/>
</owl:Class>
```

Fonte: Adaptado de PICKLER, 2014, p.95

Desta forma o sistema faz a primeira relação, mostrada na figura por uma seta saindo da elipse com a identificação “*#Bacterial_Disease*” e chegando a classe de mesmo nome, também representada por uma elipse. Este é o primeiro nível de extensividade, sendo que agora o sistema utiliza o filtro de idioma para buscar nos rótulos da nova classe se existe um termo

em português, a busca é satisfeita pelo termo “Doença bacteriana”, que é mostrado com um retângulo ao seu redor, sendo este o primeiro resultado a ser apresentado como resultado.

A etapa seguinte consiste em realizar o mesmo processo, no entanto, agora a classe em questão é “*Bacterial_Disease*”, que é uma subclasse de “*Infectious_Disease*”. O sistema então faz a segunda relação e encontra na nova classe o termo “Doença infecciosa”, mostrado também por um retângulo na figura. Neste ponto o processamento é finalizado e os resultados são apresentados.

Na Figura 31 observa-se a interface do sistema, com o formulário de entrada dos parâmetros, com o termo de entrada preenchido de acordo com o exemplo da Figura 30 e com os resultados apresentados dentro de um painel com o título “Termos Genéricos”.

Figura 31 – Interface do sistema exemplificando o teste explicado na Figura 30

Indexação Automática baseada em Ontologias

[Início](#) / Indexação Automática baseada em Ontologias

Domínio

Termo

Idioma do termo **Níveis de exaustividade**

Tipo de busca ☒ Termos Genéricos ☐ Sinônimos ou Equivalentes ☐ Cross Language

Tétano

Termos Genéricos
Doença Bacteriana
Doença Infecciosa

Fonte: desenvolvido pelo autor

No processamento o protótipo cria internamente uma consulta SPARQL de forma automática, a partir dos parâmetros “Termo”, “Idioma do termo”, “Níveis de exaustividade” e “Tipo de busca” uma consulta é formulada e aplicada a ontologia de domínio selecionada pelo parâmetro “Domínio”.

Para auxiliar na compreensão do assunto é mostrada na Figura 32 a consulta SPARQL que o sistema cria, podemos observar que o termo original “Tétano” é utilizado como ponto de partida. Um destaque também é o uso da cláusula *Filter* que recebe um parâmetro do idioma ao qual queremos efetuar a busca, pois neste exemplo partimos de uma situação hipotética onde sabe-se que o idioma do texto é o português, logo, buscam-se apenas rótulos em português.

Figura 32 – Consulta SPARQL para o exemplo de atribuição de conceitos

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (str(?label) as ?Termos)

WHERE
{
  {
    ?res1 rdfs:label "Tétano"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:label ?label
    FILTER(lang(?label) = "pt") .
  }
  UNION
  {
    ?res1 rdfs:label "Tétano"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:subClassOf ?res3 .
    ?res3 rdfs:label ?label
    FILTER(lang(?label) = "pt") . |
  }
}
```

Fonte: desenvolvido pelo autor

Ao observar o resultado apresentado na Figura 31 pode-se afirmar que é possível que um sistema realize inferências automáticas utilizando uma ontologia de domínio como vocabulário controlado, tal como é teoricamente proposto por Pickler (2014) e implementado neste protótipo.

Desta forma a proposta indexação por atribuição automática é factível utilizando este método, no entanto, é importante frisar que a quantidade de níveis hierárquicos que serão buscados para agregar termos ao documento original é variável e deve ser definida pelas políticas de indexação, pois na indexação de um material informacional quanto maior a quantidade de termos e mais genéricos estes, maior será a revocação, por outro lado menor será a precisão do sistema. Assim é imprescindível que as políticas de indexação de posse das

necessidades dos usuários do sistema de recuperação de informação definam qual a quantidade ideal de níveis hierárquicos a serem inferidos.

6.2 Descoberta de termos sinônimos

A descoberta de termos sinônimos explora uma liberdade proporcionada pela linguagem OWL, pois esta não impõe restrições sobre a utilização da propriedade *label*. De acordo com a sintaxe da OWL o uso de rótulos (*labels*) é opcional, porém caso seja utilizado podem ser declarados quantos forem necessários, principalmente para o caso de sinônimos de um mesmo conceito. Além disso por meio do uso do parâmetro *lang* pode-se distinguir qual o idioma de cada rótulo.

Utilizando este recurso da OWL é possível criar uma forma de resolver problemas de termos sinônimos, definindo vários rótulos para uma mesma entidade sendo que a indexação poderia ser feita com todas as formas de se representar uma mesma entidade do mundo real.

Na Figura 33 é possível observar um exemplo de como é feita a inferência de forma automática dentro do sistema. Assim como no exemplo de atribuição de conceitos da Figura 30 há na parte superior esquerda a representação de um documento hipotético de onde foi extraído o termo “Varicela”, mostrado logo abaixo da pequena figura. Há também a sigla “pt” que informa que este texto está no idioma português.

O restante da figura mostra outro trecho da ontologia PedTerm, que foi adaptada para este trabalho, onde é possível ver a classe com a identificação (ID) “*Chicken_Pox*”, esta classe possui dois rótulos no idioma português para demonstrar que o conceito pode ser conhecido tanto por “Varicela” quanto por “Catapora”. No mesmo exemplo é apresentado novamente o funcionamento do sistema para identificar termos hierarquicamente mais genéricos, utilizando dois níveis de exaustividade é mostrado como é o processo de inferir automaticamente os termos “Doença Viral da infância” e “Doenças ou Transtornos Pediátricos” como termos genéricos de Varicela e ainda “Catapora” como um sinônimo.

Esta abordagem de encontrar termos sinônimos é útil para que um documento que ter o termo “Varicela” possa ser indexado por “Varicela” e por “Catapora”, mesmo que o segundo não apareça na superfície textual do documento. Isto pode auxiliar na desambiguação automática, um problema de difícil resolução em sistemas de indexação automática.

Figura 33 – Exemplo de atribuição de termos sinônimos



Fonte: Adaptado de PICKLER, 2014, p.96

A consulta SPARQL apresentada na Figura 34 mostra o código necessário a busca de sinônimos na ontologia utilizando a abordagem de vários rótulos para uma mesma classe.

Figura 34 – Consulta SPARQL para o exemplo de descoberta de sinônimos

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (str(?label) as ?Termos)

WHERE
{
  {
    ?res1 rdfs:label "Varicela"@pt .
    ?res1 rdfs:label ?label |
    FILTER(lang(?label) = "pt") .
  }
}

```

Fonte: desenvolvido pelo autor

Na sequência pode-se ver a Figura 35 que apresenta o código necessário para buscar classes genéricas utilizando a mesma abordagem apresentada na subseção 6.1. O objetivo desta

figura é mostrar como o código SPARQL é idêntico alterando-se apenas o termo inicial, que agora é “Varicela”.

Figura 35 – Consulta SPARQL para o exemplo de atribuição de conceitos

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (str(?label) as ?Termos)

WHERE
{
  {
    ?res1 rdfs:label "Varicela"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:label ?label
    FILTER(lang(?label) = "pt") .
  }
  UNION
  {
    ?res1 rdfs:label "Varicela"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:subClassOf ?res3 .
    ?res3 rdfs:label ?label
    FILTER(lang(?label) = "pt") .
  }
}
```

Fonte: desenvolvido pelo autor

A soma dos resultados dos dois códigos é apresentada na Figura 36 onde a interface do sistema com todos os parâmetros de entrada é mostrada e os resultados são apresentados logo abaixo do formulário.

Os a organização dos resultados apresentados na Figura 36 foi feita de forma a facilitar a análise e compreensão, pois são mostrados dois painéis, um com o título “Termos genéricos” e outro “Termos Sinônimos”, isto é consequência da escolha das opções de mesmo nome no campo de entrada “Tipo de busca”, desta forma para um dos três tipos de busca são apresentados resultados em painéis diferentes.

Figura 36 – Interface do sistema exemplificando o teste explicado na Figura 33

Indexação Automática baseada em Ontologias

[Início](#) / Indexação Automática baseada em Ontologias

Domínio Pediatria - Saúde e Desenvolvimento Infantil

Termo Varicela

Idioma do termo Português **Níveis de exaustividade** 2

Tipo de busca ☒ Termos Genéricos ☒ Sinônimos ou Equivalentes ☐ Cross Language

Buscar Limpar

Varicela

Termos Genéricos	Termos Sinônimos
Doença Viral da infância	Catapora
Doenças ou Transtornos Pediátricos	

Fonte: desenvolvido pelo autor

Uma outra possibilidade para busca de sinônimos utilizando ontologias é a utilização do relacionamento de equivalência de classes. Nesta abordagem ao invés de declarar vários rótulos para um mesmo conceito em uma única classe declaram-se duas ou mais classes com a propriedade *equivalentClass* que permite ao sistema inferir que duas ou mais classes com rótulos diferentes representam o mesmo conceito.

Figura 37 – Exemplo de atribuição de termos sinônimos



```

<owl:Class rdf:ID="Acne">
  <rdfs:label xml:lang="en">Acne</rdfs:label>
  <rdfs:label xml:lang="es">Acné</rdfs:label>
  <rdfs:label xml:lang="pt">Acne</rdfs:label>
</owl:Class>

<owl:Class rdf:ID="Pimple">
  <rdfs:label xml:lang="en">Pimple</rdfs:label>
  <rdfs:label xml:lang="en">Spot</rdfs:label>
  <rdfs:label xml:lang="es">Espinilla</rdfs:label>
  <rdfs:label xml:lang="pt">Espinha</rdfs:label>

  <owl:equivalentClass rdf:resource="#Acne">
</owl:Class>

```

Espinha
Acne

Fonte: Adaptado de PICKLER, 2014, p.97

O exemplo mostrado da Figura 37 assim como os anteriores possui a representação de um documento no canto superior esquerdo, de onde foi extraído o termo “Espinha”. Este documento é do idioma português, pois possui a sigla “pt” e no restante da figura pode-se ver a declaração de duas classes “Acne” e “Pimple”.

O objetivo deste exemplo é demonstrar que a classe “Pimple” que possui o rótulo “Espinha” é equivalente a classe “Acne”, sendo esta uma abordagem tecnicamente diferente da apresentada na Figura 33 mas com resultados e objetivos semelhantes, já que a proposta aqui é encontrar termos sinônimos ou equivalentes.

Para o teste dessa proposta utilizam-se os mesmos parâmetros utilizados anteriormente, diferindo apenas no termo de entrada, que neste caso é “Espinha”. Os resultados da execução podem ser vistos na Figura 38.

Figura 38 – Interface do sistema exemplificando o teste explicado na Figura 37

Indexação Automática baseada em Ontologias

Início / Indexação Automática baseada em Ontologias

Domínio Pediatría - Saúde e Desenvolvimento Infantil

Termo Espinha

Idioma do termo Português **Níveis de exaustividade** 2

Tipo de busca ☒ Termos Genéricos ☒ Sinônimos ou Equivalentes ☐ Cross Language

Buscar **Limpar**

Espinha

Não foram encontrados termos genéricos.

Termos Equivalentes

Acne

Fonte: desenvolvido pelo autor

Um detalhe que pode ser observado pela primeira vez na Figura 38 é uma mensagem informando que “Não foram encontrados termos genéricos”, isto se dá devido a opção “Termos genéricos” estar marcada no campo “Tipo de busca”, sendo assim o sistema tentou buscar termos hierarquicamente genéricos, no entanto não encontrou nenhum. Logo a direita desta

mensagem é possível ver um painel intitulado “Termos equivalentes” que apresenta o termo “Acne” como termo equivalente a “Espinha”.

Ao analisar apenas os resultados apresentados nas Figuras Figura 36 e Figura 38 pode-se concluir que mesmo utilizando estruturas computacionais diferentes os resultados para fins de indexação e recuperação são os mesmos, devido a isso ambas as abordagens foram agregadas em uma única opção no sistema “Sinônimos ou Equivalentes” como um tipo de busca, sendo que em um sistema de indexação automática real ambas as abordagens seriam utilizadas para a finalidade de inferir termos sinônimos ou equivalentes nas buscas dentro do vocabulário controlado.

É importante frisar que consideramos a relação associativa de termos “sinônimos” e “equivalentes” baseado nas definições provenientes da área computacional, especificamente da linguagem OWL, que define uma equivalência entre classes como “uma construção que pode ser usada para criar classes sinônimas para ligar duas classes em uma mesma ontologia ou em ontologias diferentes” (CARVALHO; LIMA, 2005).

Outra conclusão obtida após a análise das figuras supracitadas é a possibilidade da implementação da descoberta automática de termos hierarquicamente genéricos junto da descoberta de sinônimos e/ou termos equivalentes o que indica uma hipótese promissora no processo de desambiguação de termos em sistema de indexação automática.

Figura 39 – Consulta SPARQL para o exemplo descoberta de termos equivalentes

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>

SELECT (str(?label) as ?Termos)
WHERE
{
  {
    ?res1 rdfs:label "Espinha"@pt .
    ?res1 owl:equivalentClass ?res2 .
    ?res2 rdfs:label ?label
    FILTER(lang(?label) = "pt") .
  }
}
```

Fonte: desenvolvido pelo autor

A Figura 39 apresenta o código SPARQL responsável por permitir que o sistema seja capaz de buscar por termos equivalentes a um conceito qualquer, com pode-se observar o código difere do apresentado na Figura 34 pelo uso da propriedade “*equivalenClass*”, que é um recurso específico do subconjunto de propriedades da OWL, diferente da propriedade “*label*” que é parte do subconjunto RDFS. Observa-se devido a isso a necessidade da inclusão de um novo prefixo “*owl:*” que permite ao interpretador SPARQL utilizar as propriedades oferecidas por este subconjunto além do “*rdfs:*”.

Por fim conclui-se que a proposta de descoberta de termos sinônimos e/ou equivalentes é válida e tecnicamente factível, como a teoria proposta afirma, sendo este mais um possível recurso a ser agregado aos sistemas de indexação automática visando melhorias no processo. No entanto para que este método funcione é necessário que a ontologia seja construída com todas as relações de equivalência possíveis, pois caso relações sejam omitidas o sistema será incapaz de inferir corretamente.

A título de exemplo pode-se citar as declarações da Figura 37, onde a classe “Pimple”, que representa o termo “Espinha” possui a propriedade que declara que ela é equivalente a classe “Acne” que representa o mesmo termo, no entanto a relação inversa não é declarada, desta forma caso o termo encontrado no documento hipotético tivesse sido “Acne” o sistema não seria capaz de inferir que “Acne” é um conceito equivalente a “Espinha”, devido a classe “Acne” não possuir a propriedade equivalente indicando a classe “Pimple”.

6.3 Indexação Multilíngue

A indexação multilíngue consiste em explorar a liberdade da linguagem OWL de criar um ou mais rótulos para uma mesma classe, e, utilizando o parâmetro *xml:lang* é possível que uma mesma classe seja descrita em vários idiomas diferentes, com isso uma ontologia pode ser utilizada para indexação de um corpus contendo documentos em diferentes idiomas.

A construção de uma ontologia de domínio é uma tarefa custosa, pois envolve investimentos de tempo e dinheiro no sentido de que é fortemente recomendado que essa tarefa seja feita por profissionais capacitados para tal finalidade, desta forma a possibilidade de se utilizar uma ontologia independente de seu idioma pode propiciar a universalização das ontologias, pois é possível utilizar por exemplo uma ontologia construída no idioma inglês para

indexar documentos em português, sendo necessário apenas criar em todas as classes os rótulos identificados com o parâmetro *lang* igual a português (“pt”).

A tarefa de “traduzir” uma ontologia, ou mais precisamente, traduzir os rótulos de uma ontologia para outros idiomas permite que uma ontologia criada seja única, sendo que para necessidades mais específicas, ontologias de tarefa podem ser criadas como ferramentas auxiliares a uma ontologia mais genérica.

Neste trabalho foi utilizada uma ontologia construída no idioma inglês, como exemplificado nos parágrafos anteriores. Foi necessário criar os rótulos como descrito, porém esta é uma tarefa consideravelmente mais simples do que recriar toda uma ontologia.

O exemplo apresentado na Figura 40 mostra uma abordagem bastante semelhante as propostas de atribuição de conceitos hierarquicamente genéricos, diferindo agora no idioma de entrada do termo e também nos resultados apresentados pelo sistema.

A Figura 40 apresenta na lateral esquerda três pequenas imagens que simbolizam três documentos hipotéticos, cada um em um idioma distinto. O primeiro em inglês “en”, o segundo em espanhol “es” e o terceiro em português “pt”. Logo abaixo de cada pequena imagem é mostrada uma palavra escrita em cor preta que representa um termo extraído deste documento e, abaixo desta, dois outros termos obtidos por inferência do sistema utilizando o conceito de atribuição automática de termos hierarquicamente genéricos apresentado na subseção 6.1. As cores de cada um destes termos obtidos por inferência auxiliam a compreensão, pois os termos em azul (idioma inglês) foram submetidos ao sistema com o filtro de idioma igual a inglês, os dois outros termos seguem a mesma regra, cada um com seu respectivo idioma.

Figura 40 – Exemplo de Indexação multilíngue



Fonte: Adaptado de PICKLER, 2014, p.98

O resultado da execução deste teste juntamente com os parâmetros de entrada no sistema pode ser visto na Figura 41, onde destaca-se que o termo informado como entrada está no idioma inglês, conseqüentemente é necessário escolher essa opção no campo “Idioma do termo”, o que resulta em uma resposta do sistema que apresenta dois termos hierarquicamente genéricos no idioma inglês.

Figura 41 – Interface do sistema exemplificando o teste explicado na Figura 40

Início / Indexação Automática baseada em Ontologias

Domínio
Pediatria - Saúde e Desenvolvimento Infantil

Termo
Hypothyroidism

Idioma do termo
Inglês

Níveis de exaustividade
2

Tipo de busca
☒ Termos Genéricos
☐ Sinônimos ou Equivalentes
☐ Cross Language

Buscar
Limpar

Hypothyroidism

Termos Genéricos

Endocrine System Disorder

Disease or Disorder

Fonte: desenvolvido pelo autor

A mesma lógica se aplica a Figura 42, onde é apresentado um teste para descoberta de termos hierarquicamente genéricos a partir da entrada de um termo no idioma espanhol.

Figura 42 – Interface do sistema exemplificando o teste explicado na Figura 40

Início / Indexação Automática baseada em Ontologias

Domínio
Pediatria - Saúde e Desenvolvimento Infantil

Termo
Hipotiroidismo

Idioma do termo
Espanhol

Níveis de exaustividade
2

Tipo de busca
☒ Termos Genéricos
☐ Sinônimos ou Equivalentes
☐ Cross Language

Buscar
Limpar

Hipotiroidismo

Termos Genéricos

Trastorno del Sistema Endocrino

Emfermedad o Trastorno

Fonte: desenvolvido pelo autor

Por último, na Figura 43 pode-se observar a simulação da descoberta de termos a partir de um termo no idioma português, sendo possível verificar a capacidade do sistema em operar com uma única ontologia em diferentes idiomas, isto se dá devido a declaração de rótulos nestes três idiomas.

Figura 43 – Interface do sistema exemplificando o teste explicado na Figura 40

The screenshot shows a web interface for a system titled 'Início / Indexação Automática baseada em Ontologias'. The search parameters are as follows:

- Domínio:** Pediatric - Health and Child Development (Pediatria - Saúde e Desenvolvimento Infantil)
- Termo:** Hypothyroidism (Hipotireoidismo)
- Idioma do termo:** Portuguese (Português)
- Níveis de exaustividade:** 2
- Tipo de busca:** ☒ Terms Genéricos, ☐ Synonyms or Equivalents, ☐ Cross Language

Buttons for 'Buscar' (Search) and 'Limpar' (Clear) are present. Below the search parameters, the results for 'Hipotireoidismo' are displayed in a table:

Termos Genéricos
Transtorno do Sistema Endócrino
Doença ou Distúrbio

Fonte: desenvolvido pelo autor

A consulta SPARQL gerada pelo sistema para se executar a proposta exemplificada na Figura 43 pode ser vista na Figura 44. Destaca-se que a única diferença da consulta apresentada na Figura 32, referente a descoberta de termos hierarquicamente genéricos, é o termo que foi informado como entrada. A mesma coisa acontece com as consultas geradas para produzir os resultados apresentados nas Figuras Figura 41 e Figura 42 e Figura 43, por esta razão estas consultas não são mostradas.

Figura 44 – Consulta SPARQL para o exemplo de indexação multilíngue

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (str(?label) as ?Termos)
WHERE
{
  {
    ?res1 rdfs:label "Hipotireoidismo"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:label ?label
    FILTER(lang(?label) = "pt") .
  }
  UNION
  {
    ?res1 rdfs:label "Hipotireoidismo"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:subClassOf ?res3 .
    ?res3 rdfs:label ?label
    FILTER(lang(?label) = "pt") .
  }
}
```

Fonte: desenvolvido pelo autor

As explicações sobre a indexação multilíngue apresentadas até aqui são necessárias, porém, pouco relevantes do ponto de vista da evolução dos métodos de indexação automática.

A proposta de indexação multilíngue torna-se mais interessante quando utilizamos uma outra abordagem, considerando um documento hipotético no idioma português onde tenha sido extraído o “Hipotireoidismo” submete-se este termo ao sistema para que seja feita uma busca automática por termos hierarquicamente mais genéricos permitindo assim a indexação deste documento por um número maior de descritores de assunto visando maximizar seu potencial de recuperação.

No entanto, utilizando a abordagem multilíngue é possível indexar este documento com o termo “Hipotireoidismo” em todos os idiomas contemplados pela ontologia em questão, neste caso Espanhol e Inglês. Essa abordagem é apresentada na Figura 45, onde pode-se ver como resultado da execução do teste o primeiro painel com os dois termos hierarquicamente genéricos de “Hipotireoidismo” e um segundo painel intitulado “*Cross Language*” com os termos “Hipotireoidismo”, “Doença ou Distúrbio” e Transtorno do Sistema Endócrino” em inglês e em espanhol respectivamente.

Isto é possível pois foi selecionada a opção “*Cross Language*” no campo tipo de busca, o que internamente no sistema faz com que além de fazer a busca por termos genéricos retorna os termos relacionados também nos demais idiomas contemplados pela ontologia.

Figura 45 – Interface do sistema exemplificando o teste explicado na Figura 40

Indexação Automática baseada em Ontologias

Início / Indexação Automática baseada em Ontologias

Domínio

Pediatria - Saúde e Desenvolvimento Infantil

Termo

Hipotireoidismo

Idioma do termo

Português

Níveis de exaustividade

2

Tipo de busca

☒ Termos Genéricos
 ☐ Sinônimos ou Equivalentes
 ☒ *Cross Language*

Buscar

Limpar

Hipotireoidismo

Termos Genéricos	<i>Cross Language</i>
Transtorno do Sistema Endócrino	(Inglês) Disease or Disorder
	(Inglês) Endocrine System Disorder
	(Inglês) Hypothyroidism
	(Espanhol) Enfermedad o Trastorno
	(Espanhol) Hipotiroidismo
	(Espanhol) Trastorno del Sistema Endocrino

Fonte: desenvolvido pelo autor

Para efeitos de codificação SPARQL essa abordagem é bastante simples, pois o sistema utiliza uma consulta padrão para buscar conceitos hierarquicamente genéricos tal como as apresentas nas Figuras Figura 32 e Figura 35 diferindo apenas na supressão da cláusula “*FILTER*”, responsável por filtrar os resultados apenas para o idioma especificado no campo “Idioma do termo” presente no formulário de entrada de dados. Na Figura 46 pode-se ver a consulta gerada pelo sistema para produzir o resultado apresentado na Figura 45.

Figura 46 – Consulta SPARQL para o exemplo de indexação multilíngue

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (?label as ?Termos) (lang(?label) as ?idioma)
WHERE
{
  {
    ?res1 rdfs:label "Hipotireoidismo"@pt .
    ?res1 rdfs:label ?label .
  }
  UNION
  {
    ?res1 rdfs:label "Hipotireoidismo"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:label ?label .
  }
  UNION
  {
    ?res1 rdfs:label "Hipotireoidismo"@pt .
    ?res1 rdfs:subClassOf ?res2 .
    ?res2 rdfs:subClassOf ?res3 .
    ?res3 rdfs:label ?label .
  }
}
ORDER BY ?idioma
```

Fonte: desenvolvido pelo autor

Desta forma conclui-se que é possível agregar aos sistemas de indexação automática a capacidade de representar os documentos de um acervo em diferentes idiomas, o limite de idiomas para a indexação é fixado pela ontologia, pois para que o sistema possa utilizar determinado idioma, este precisa estar especificado nas classes da ontologia utilizando rótulos (*labels*) com o parâmetro *lang* igual a sigla do idioma (“pt”, “en”, “es”, “fr”, etc).

Esta é uma abordagem que pode auxiliar na indexação de documentos de algumas áreas técnicas como a computação, onde vários termos candidatos a descritores dos documentos são em inglês, mesmo o documento ser do idioma português, o que possivelmente iria gerar uma inconsistência caso o vocabulário controlado estivesse representado apenas no idioma do documento.

Considerações finais

As pesquisas em sistemas de indexação automática têm por objetivo tornar o processo de representação de documentos mais ágil, pois, considerando a quantidade de documentos disponíveis é possível compreender a grande demanda por mecanismos que permitam a automatização deste processo.

O processo de indexação é uma tarefa que lida com elementos de natureza linguística, sofrendo influência de fatores subjetivos e de difícil formalização. Neste contexto, o uso de vocabulários controlados possibilita a obtenção de melhores resultados.

Assim como na indexação manual, a indexação automática também apresenta benefícios com a utilização de vocabulários controlados. No entanto, é necessário buscar opções de vocabulários que sejam adequados ao uso por sistemas computacionais, tal como as ontologias.

A utilização de ontologias vem de encontro a necessidade de vocabulários controlados processáveis por máquinas, tal como afirma Narukawa (2011), a adoção de ontologias em substituição de outros vocabulários, como tesauros, em sistemas de indexação automática pode prover mais confiabilidade ao sistema, pois estas podem prover um tratamento semântico das informações.

Para que uma ontologia seja utilizada como um vocabulário controlado, é necessário que ela seja construída para esta finalidade, ou, no caso do uso de ontologias previamente elaboradas, há a necessidade de ajustes, lembrando que, para a proposta apresentada por esta pesquisa a declaração dos rótulos dos conceitos (*labels*) com seus respectivos idiomas torna-se obrigatória.

Os resultados obtidos a partir do protótipo desenvolvido no contexto deste trabalho permitem concluir que, o uso de ontologias como ferramenta de controle terminológico em

sistemas de indexação automática pode aprimorar o processo de indexação, implicando em melhoras na recuperação de informação como um todo.

Sendo o objetivo desta pesquisa propor e desenvolver um método de utilização de ontologias em uma das etapas do processo de indexação automática de documentos textuais, é possível afirmar, a partir dos resultados obtidos, que a proposta é factível e aplicável, especificamente junto a etapa posterior a extração dos termos candidatos a descritores fornecendo meios para análise e padronização do índice.

Destaca-se a importância da implementação do protótipo, pois, por meio das pesquisas para seu desenvolvimento foi possível observar a importância da utilização das tecnologias e propostas da web semântica no desenvolvimento de sistemas de informação e comunicação mais inteligentes.

A partir das leituras e comparações com outros trabalhos correlatos foi possível observar que há muitos autores discutindo sobre o uso de ontologias em processos de indexação, mas poucos apresentam propostas concretas ou resultados sólidos da real aplicação dessas propostas.

Alguns trabalhos, de áreas como a computação, possuem um enfoque mais prático, com a proposta de criar sistemas utilizando ontologias em seu desenvolvimento, mas, sem explorar em profundidade temas como a importância do processo de indexação e da definição das políticas de indexação, fatores estes que influenciam o funcionamento dos sistemas de recuperação de informação.

Por fim, este trabalho busca ser uma colaboração para os estudos em indexação automática utilizando ontologias como vocabulários controlados, pois apresenta, além dos resultados obtidos, uma metodologia bastante clara de como toda a implementação foi feita, visando possibilitar que trabalhos futuros se utilizem dos conhecimentos apresentados para desenvolver novas e mais eficientes técnicas que possam cada vez mais melhorar a indexação e, conseqüentemente, a busca por informações relevantes.

REFERÊNCIAS

AIRIO, E.; JÄRVELIN, K.; SAATSI, P.; KEKÄLÄINEN, J.; SUOMELA, S. CIRI – na ontologybased query interface for text retrieval. In: HYVÖNEN, E.; KAUPPINEN T.; SALMINEN, M.; VILJANEN, L.; ALASIURU, P. (Eds) **Proceedings of the 11th Finnish Artificial Intelligence Conference**, 2004.

ALBRECHTSEN, Hanne. Subject analysis and indexing: from automated indexing to domain analysis. **The Indexer**, v.18, n.4, p.219-224, out. 1993. Disponível em: <http://www.theindexer.org/files/18-4/18-4_219.pdf>. Acesso em: 29 Fev. 2016.

AMORIM, S. R. L.; CHERIAF, Malik. **Sistema de Indexação e Recuperação de Informação em Construção Baseado em Ontologia**. In: III Encontro Tecnologia da Informação e Comunicação na Construção Civil - TIC2007, Porto Alegre, Brasil, 2007.

AMORIM, S. R. L.; RABELO, P. F. R. **ONTOARQ – Ontologia para Arquitetura, Engenharia e Construção. Visualização e Gerenciamento na Web**. In: III Encontro Tecnologia da Informação e Comunicação na Construção Civil - TIC2007, Porto Alegre, Brasil, 2007.

ARAÚJO, Carlos Alberto Ávila. O que é Ciência da Informação?. **Informação & Informação**, [S.l.], v. 19, n. 1, p. 01-30, dez. 2013. ISSN 1981-8920. Disponível em: <<http://www.uel.br/revistas/uel/index.php/informacao/article/view/15958>>. Acesso em: 04 Ago. 2015. doi:10.5433/1981-8920.2014v19n1p01.

BORGES, Graciane Silva Bruzinga; MACULAN, Benildes Coura Moreira dos Santos; LIMA, Gercina Angela Borem de Oliveira. Indexação automática e semântica: estudo da análise do conteúdo de teses e dissertações. **Informação & Sociedade: Estudos**, v. 18, n. 2, p.181-193, 2008. Disponível em: <<http://www.ies.ufpb.br/ojs2/index.php/ies/article/download/1759/2129>>. Acesso em: 06 abr. 2016.

BORGES, Graciane S. B. **Indexação automática de documentos textuais: critérios essenciais**. 2009. 111f. Dissertação (Mestrado em Ciência da Informação) - Escola de Ciência da Informação, Universidade Federal de Minas Gerais. Belo Horizonte, 2009.

BORKO, H. Toward a theory of indexing. *Information Processing and Management*, v.13, p. 355-365, 1977 apud GUEDES, Vânia L. S. Estudo de um critério para indexação automática derivativa de textos científicos e tecnológicos. **Ciência da Informação**, Brasília, DF, v. 23, n. 3, p.318-326, set./dez. 1994.

BRÄSCHER, M. A ambigüidade na recuperação da informação. **DataGramaZero: Revista de Ciência da Informação**, v. 3, n. 1, fev. 2002. Disponível em: <http://repositorio.unb.br/bitstream/10482/924/2/ARTIGO_AmbiguidadeRecuperacaoInformacao.pdf>. Acesso em: 24 de mar. 2016.

BUCKLAND, M.K. Information as thing, **Journal of the American Society of Information Science**, v.42, n.5, 1991. p.351-360.

CARNEIRO, M. U. Avaliação de desempenho de pessoal em bibliotecas: uma revisão de literatura. **Ci. Inf.**,Brasília, v.14, n.1, p. 25-35, 1985.

DE LIMA, Júnio César; DE CARVALHO, Cedric L. **Ontologias-owl (web ontology language)**. Technical report, Universidade Federal de Goiás, 2005.

CASTELLS, Pablo; FERNANDEZ, Miriam; VALLET, David. An Adaptation of the Vector-Space Model for Ontology-Based Information Retrieval. **Ieee Trans. Knowl. Data Eng.**, [s.l.], v. 19, n. 2, p.261-272, fev. 2007. Institute of Electrical & Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/tkde.2007.22>.

CESARINO, Maria Augusta da Nóbrega; PINTO, Maria Cristina Mello Ferreira. Sistemas de recuperação da informação. **Revista da Escola de Biblioteconomia da UFMG**. Belo Horizonte, v. 14, n. 2, p. 157-168, 1985.

CHANDRASEKARAN, B.; JOSEPHSON, J.R.; BENJAMINS, V. R.1999. What are ontologies, and why do we need them ? **IEEE Intelligent Systems**, v.14, n.1, 1999.

CHAUMIER, Jacques. Indexação: conceito, etapas e instrumentos. **Revista Brasileira de Biblioteconomia e Documentação**, São Paulo, v. 21, n. 1/2, p. 63-79, 1988. apud GIL LEIVA, I. **Manual de indización: teoría y práctica**. Gijón: Trea, 2008. 429 p.

CUNHA, Isabel M. R. Ferin - “Análise documentária”. In: SMIT, Johanna Wilhelmina (Coord.) - **Análise documentária: a análise da síntese**. 2.^a ed. Brasília: IBICT, 1989. p. 39-62.

DUCHARME, Bob. **Learning SPARQL: Querying and Updating with SPARQL 1.1**. 2. ed. Sebastopol (USA): O' Reilly, 2013.

FERREIRA, Sueli Mara Soares Pinto; PINTO, Soares. Novos paradigmas e novos usuários de informação. **Ciência da Informação**, v. 25, n. 2, p. 1-10, 1995.

FERNEDA, Edberto. **Modelos computacionais de recuperação de informação**. Rio de Janeiro: Ciência Moderna, 2012.

_____. **Ontologia como recurso de padronização terminológica em um Sistema de Recuperação de Informação**. 2013. 109 f. Relatório (Estágio Pós-Doutoral), Ciência da Informação, Universidade Federal da Paraíba, João Pessoa. 2013.

FIDEL, Raya. User-centered indexing. **Journal of the American Society for Information Science** (1986-1998), v. 45, n. 8, p. 572, 1994.

FILHO, F.W.; LÓSCIO, B.F. Web Semântica: Conceitos e Tecnologias. In: Pedro de Alcantara. (Org.). **II Escola Regional de Computação** - Ceará, Maranhão e Piauí - ERCEMAPI 2009. : SBC, 2009.

FUJITA, Mariângela Spotti Lopes. Avaliação da eficácia de recuperação do sistema de indexação PRECIS. **Ciência da Informação**, [S.l.], v. 18, n. 2, Dez. 1989. ISSN 1518-8353. Disponível em: <<http://revista.ibict.br/ciinf/index.php/ciinf/article/view/1361/987>>. Acesso em: 07 Set. 2015.

_____, Mariângela Spotti Lopes. (2012). A política de indexação para representação e recuperação da informação. In: I. Gil Leiva & M. S. L. Fujita (Ed.). **Política de indexação** (p. 17-28). São Paulo: Cultura Acadêmica.

_____, Mariângela Spotti Lopes. A leitura do indexador: estudo de observação. **Perspectivas em Ciência da Informação**, Belo Horizonte, v. 4, n. 1, p. 101-116, 1999.

_____, Mariângela Spotti Lopes. **Leitura em análise documentária**. Marília: UNESP; CNPq, 1999. Relatório Parcial de Pesquisa.

GARFIELD, Eugene. A Tribute To Calvin N. Mooers, A Pioneer Of Information Retrieval. **The Scientist**, [s. L.], v. 11, n. 4, p.1-3, 17 mar. 1997.

GIL LEIVA, I. **Manual de indización: teoría y práctica**. Gijón: Trea, 2008. 429 p.

GIL LEIVA, I. Aspectos conceituais da indexação. In: FUJITA, Mariângela Spotti Lopes; LEIVA, Isidoro Gil. **Política de Indexação**. Marília: Cultura Acadêmica, 2012. p. 31-106.

GIL LEIVA, I, RUBI, M. P., FUJITA, M. S. L. Consistência na indexação em bibliotecas universitárias brasileiras. **Transinformação**, Campinas, v. 20, p. 233-54, 2008.

GRUBER, T. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. **International Journal Human-Computer Studies**, v.43, n.5-6, 1995.

GUIMARÃES, F J Z. **Utilização de ontologias no domínio B2C**. 2002. 195 f. Dissertação (Mestrado em Informática), Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2002.

JASPER, R.; USCHOLD, M. A Framework for understanding and classifying ontology applications. In: **KRR5-99**, Stockholm. 1999.

KARA, Soner et al. An ontology-based retrieval system using semantic indexing. **Information Systems**, [s.l.], v. 37, n. 4, p.294-305, jun. 2012. Elsevier BV. <http://dx.doi.org/10.1016/j.is.2011.09.004>.

KEYSER, Pierre de. **Indexing: from thesauri to the semantic web**. Sawston, England: Chandos Publishing, 2012. 249 p.

KOBASHI, N. Y. **A elaboração de informações documentárias: em busca de uma metodologia**. 1994. 195f. Tese (Doutorado em Ciências da Comunicação – Escola de Comunicação e Artes, Universidade de São Paulo, São Paulo, 1994.

KURAMOTO, Hélio. Sintagmas nominais: uma nova proposta para a recuperação de informação. **DataGramaZero - Revista de Ciência da Informação**, Rio de Janeiro, v. 3, n. 1, fev. 2002. Disponível em: <http://www.dgz.org.br/fev02/Art_03.htm> Acesso em: 26 ago. 2015

LANCASTER, F.W. **Indexação e Resumos: teoria e prática**. 2.ed. Brasília, DF: Briquet de Lemos, 2004.

LE COADIC, Yves-françois. **A ciência da informação**. Tradução de Maria Yêda F.s. de Filgueiras Gomes. 2. ed. Brasília: Briquet de Lemos, 2004.

LIJUN, Tang; XU, Chen. The Study of Semantic Retrieval Based on the Ontology of Teaching Management. **Procedia Engineering**, [s.l.], v. 15, p.1555-1559, 2011. Elsevier BV. <http://dx.doi.org/10.1016/j.proeng.2011.08.289>.

LIMA, J. C.; CARVALHO, C. L. **Ontologias – OWL (Web Ontology Language)**. 2005. Relatório Técnico - Instituto de Informática Universidade Federal de Goiás. Disponível em: <http://www.portal.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_004-05.pdf>. Acesso em: 28 set. 2015.

LOPES, Ilza Leite. Uso das linguagens controlada e natural em bases de dados: revisão da literatura. **Ciência da informação**, v. 31, n. 1, p. 41-52, 2002.

MAIMONE, G. D.; SILVEIRA, N. C.; TÁLAMO, M. de. F. G. M. Reflexões acerca das relações entre representação temática e descritiva. **Informação & Sociedade: Estudos**, João

Pessoa, v.21, n.1, p. 27-35, jan./abr. 2011. Disponível em:
<<http://www.ies.ufpb.br/ojs2/index.php/ies/issue/archive>. Acesso em: 08 fev. 2016

MANAF, Nor Azlinayati Abdul; BECHHOFFER, Sean; STEVENS, Robert. A Survey of Identifiers and Labels in OWL Ontologies. **Proceedings of the 6th International Workshop on OWL Experiences and Directions (OWLED)**, 2010.

MANNING, Christopher D.; RAGHAVAN, Prabhakar; SCHÜTZE, Hinrich. **An Introduction to Information Retrieval**. Cambridge, England: Cambridge University Press, 2009. 581 p.

MARCONDES, C., ALMEIDA CAMPOS, M. Ontologia e web semântica: o espaço da pesquisa em Ciência da Informação. **Ponto de Acesso**, Salvador, v.2, n.1, p.107-136, 2008. Disponível em: <http://www.portalseer.ufba.br/index.php/revistaici/article/view/2669/1885>. Acesso em: 28 Set. 2015.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos da metodologia científica**. 7. ed. São Paulo: Atlas, 2010, 297 p.

MCGUINNESS, Deborah L. Ontologies Come of Age. In: FENSEL, Dieter et al. **Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential**. Cambridge, England: Mit Press, 2003. p. 171-222. Disponível em: <[http://www-ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-\(with-citation\).htm](http://www-ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-(with-citation).htm)>. Acesso em: 24 set. 2015.

MOOERS, C. Zatocoding applied to mechanical organization of knowledge. **American Documentation**, v. 2, n. 1, 1951, p.20-32.

MORRIS, Ruth CT. Toward a user-centered information service. **Journal of the American Society for Information Science** (1986-1998), v. 45, n. 1, p. 20, 1994.

NARUKAWA, C. Miyuki. **Estudo de vocabulário controlado na indexação automática: aplicação no processo de indexação do Sistema de Indización Semiautomática (SISA)**. 2011. 222 f. Dissertação (mestrado) - Universidade Estadual Paulista, Faculdade de Filosofia e Ciências, 2011.

NOVELLINO, Maria Salet Ferreira. Instrumentos e metodologias de representação da informação. **Informação & Informação**, [S.l.], v. 1, n. 2, p. 37-45, dez. 1996. ISSN 1981-8920. Disponível em:
<<http://www.uel.br/revistas/uel/index.php/informacao/article/view/1603>>. Acesso em: 21 Ago. 2015. doi:10.5433/1981-8920.1996v1n2p37.

PÉREZ, Jorge; ARENAS, Marcelo; GUTIERREZ, Claudio. Semantics and Complexity of SPARQL. In: **International semantic web conference**. 2006. p. 30-43.

PICKLER, Maria Elisa Valentim. **Diretrizes para utilização de ontologias na indexação automática**. 2014. 101f. Dissertação (Mestrado em Ciência da Informação) – Faculdade de Filosofia e Ciências, Universidade Estadual Paulista, Marília, 2014.

_____, Maria Elisa Valentim. Web Semântica: ontologias como ferramentas de representação do conhecimento. **Perspect. ciênc. inf.**, abr. 2007, v.12, n.1, p.65-83.

_____, Maria Elisa Valentim. **A Web Semântica: ontologia ou tesouro?** Uma revisão de literatura. 2006. 82 f. Trabalho de Conclusão de Curso (Graduação em Biblioteconomia) - Universidade Estadual de Londrina, Londrina, 2006.

PRUD'HOMMEAUX, Eric et al. SPARQL query language for RDF. **W3C recommendation**, v. 15, 2008.

RAMALHO, R.A.S. **Web semântica: aspectos interdisciplinares da gestão de recursos informacionais no âmbito da ciência da informação**. 2006. 120 f. Dissertação (mestrado) - Universidade Estadual Paulista, Faculdade de Filosofia e Ciências, 2006.

RAMALHO, R.A.S. **Desenvolvimento e utilização de ontologias em bibliotecas digitais: uma proposta de aplicação**. 2010. 145 f. Tese (Doutorado) - Universidade Estadual Paulista, Faculdade de Filosofia e Ciências, 2010.

RIECKEN, R. F. Frame de temas potenciais de pesquisa em ciência da informação. **Revista Digital de Biblioteconomia e Ciência da Informação**, Campinas, v. 3, n. 2, p. 43-63, jan./jun., 2006.

RUBI, Milena Polsinelli. **Política de indexação para construção de catálogos coletivos em bibliotecas universitárias**. 2008. 166 f. Tese (Doutorado) - Universidade Estadual Paulista, Faculdade de Filosofia e Ciências de Marília, 2008.

SANTAREM SEGUNDO, J E. **Representação Iterativa: um modelo para repositórios digitais**. 2010. 224f. Tese (Doutorado em Ciência da Informação) – Faculdade de Filosofia e Ciências, Universidade Estadual Paulista, Marília, 2010.

SARACEVIC, Tefko. Ciência da informação: origem, evolução e relações. **Perspectivas em Ciência da Informação**, Belo Horizonte, v. 1, n. 1, p. 41-62, jan./jun. 1996.

SEVERINO, Antônio Joaquim. **Metodologia do trabalho científico**. 23. ed. rev. atual. São Paulo: Cortez, 2007.

SHANNON, C.; WEAVER, W. **The mathematical theory of communication**. Urbana, IL: University of Illinois Press, 1972. (Original work published in 1949).

SILVA, M. dos R. da; FUJITA, M. S. L. A prática de indexação: análise evolutiva de tendências teóricas e metodológicas. **Transinformação**, Campinas, v. 16, n. 2, p.133-161, 2004.

SILVA, Renata Eleuterio da; SANTOS, Plácida Leopoldina Ventura Amorim da Costa; FERNEDA, Edberto. Modelos de recuperação de informação e web semântica: a questão da relevância. **Informação & Informação**, [S.l.], v. 18, n. 3, p. 27-44, out. 2013. ISSN 1981-8920. Disponível em: <<http://www.uel.br/revistas/uel/index.php/informacao/article/view/12822>>. Acesso em: 21 Ago. 2015. doi:10.5433/1981-8920.2013v18n3p27.

SOERGEL, D. The rise of ontologies or the reinvention of classification. **Journal of the American Society for Information Science**. v. 50, n. 12, 1999.

SOUZA, Renato Rocha. Sistemas de recuperação de informações e mecanismos de busca na web: panorama atual e tendências. **Perspectivas em Ciência da Informação**, Belo Horizonte, v. 11, n. 2, dez. 2006. p. 161-173. Disponível em: <<http://www.scielo.br/pdf/pci/v11n2/v11n2a02.pdf>> Acesso em: 26 ago. 2015

UNISIST. Princípios de indexação. **Revista da Escola de Biblioteconomia da UFMG**, Belo Horizonte, v.10, n.1, p. 83-94, mar. 1981.

VICKERY, Brian C. Ontologies. **Journal of information science**, v. 23, n. 4, p. 277-286, 1997.

VIEIRA, Simone Bastos. **Rede SABI**: diretrizes de indexação. Brasília: Senado Federal, 1996. 57p.

Whetzel PL, Noy NF, Shah NH, Alexander PR, Nyulas C, Tudorache T, Musen MA. **BioPortal**: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic Acids Res.* 2011 Jul; 39(Web Server issue):W541-5. Epub 2011 Jun 14.