

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
INFORMÁTICA INDUSTRIAL

ALBERTO HEITOR MOLINARI

**INDEXAÇÃO DE ACÓRDÃOS POR MEIO DE UMA ONTOLOGIA
JURISPRUDENCIAL POPULADA A PARTIR DE UM CORPUS
JURÍDICO REAL**

DISSERTAÇÃO DE MESTRADO

CURITIBA

2011

ALBERTO HEITOR MOLINARI

**INDEXAÇÃO DE ACÓRDÃOS POR MEIO DE UMA ONTOLOGIA
JURISPRUDENCIAL POPULADA A PARTIR DE UM CORPUS
JURÍDICO REAL**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de “Mestre em Ciências” – Área de Concentração: Engenharia de Automação e Sistemas.

Orientador: Prof. Cesar Augusto Tacla, Dr.

CURITIBA

2011

Dados Internacionais de Catalogação na Publicação

- M722 Molinari, Alberto Heitor
Indexação de acórdãos por meio de um ontologia jurisprudencial populada a partir de um corpus jurídico real / Alberto Heitor Molinari.— 2011.
198 p. : il. ; 30 cm
- Orientador: Cesar Augusto Tacla.
Dissertação (Mestrado) – Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial. Curitiba, 2011.
Bibliografia: p. 165-168.
1. Sistemas de recuperação da informação. 2. Jurisprudência – Pesquisa. 3. Ontologia. 4. Web semântica. 5. Indexação. 6. Engenharia elétrica – Dissertações. I. Tacla, Cesar Augusto, orient. II. Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial. III. Título.

CDD (22. ed.) 621.3

Título da Dissertação Nº 576:

**“Indexação de Acórdãos por meio de uma Ontologia
Jurisprudencial Populada a partir de um Corpus
Jurídico Real”**

por

Alberto Heitor Molinari

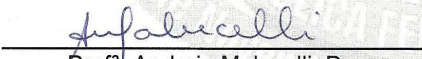
Esta dissertação foi apresentada como requisito parcial à obtenção do grau de MESTRE EM CIÊNCIAS – Área de Concentração: Engenharia de Automação e Sistemas, pelo Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial – CPGEI – da Universidade Tecnológica Federal do Paraná – UTFPR – Campus Curitiba, às 9h do dia 30 de agosto de 2011. O trabalho foi aprovado pela Banca Examinadora, composta pelos professores:



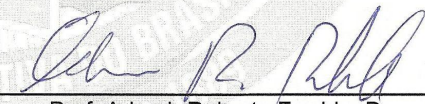
Prof. Cesar Augusto Tacla, Dr.
(Presidente – UTFPR)



Prof. Emerson Cabrera Paraiso, Dr.
(PUC-PR)



Profª Andreia Malucelli, Dr.
(PUC-PR)



Prof. Ademir Roberto Freddo, Dr.
(UTFPR - FB)

Visto da coordenação:



Prof. Fábio Kurt Schneider, Dr.
(Coordenador do CPGEI)

Dedico este trabalho à minha esposa e filhos, Tania, Caio e Davi, fontes de inspiração e perseverança, e ao meus pais, Luiz e Izabel, que me tornaram capaz de sonhar e realizar.

AGRADECIMENTOS

A Deus, fonte da vida e do conhecimento, ao qual devo todas as minhas realizações.

À minha querida família, Tania, Caio e Davi, pelo apoio e confiança incondicionais em mim depositados, tornando possível o sucesso do meu Mestrado.

Ao meus valiosos pais, Luiz e Izabel, pela vida e pelo conhecimento a mim consagrados.

Ao Prof. Cesar Augusto Tacla, PhD, por ter acreditado em mim e no meu projeto e por mostrar que um Orientador é, antes de tudo, um facilitador e companheiro.

A Denis Felipe Suzuki, MsC, pela motivação em persistir no sonho do mestrado e pela sugestão do tema.

A Maria Inês Levis Costa, diretora do Depto. de Tecnologia da Informação e Comunicação, pelo apoio e orientação dentro do TJPR.

A Alcimara, Ana Z., Denise e Rosalina, especialistas do Centro de Documentação do TJPR, que consignaram a esta dissertação todo o seu conhecimento no domínio da jurisprudência.

Aos Profs. Ademir Freddo, PhD, Andreia Malucelli, PhD, Emerson Paraíso, PhD, e Moacyr Molinari, MsC, pelas minuciosas revisões, as quais promoveram o enriquecimento do presente trabalho.

Ao colega de pós-graduação Ademir Freddo, pelas orientações na implementação de aplicações baseadas em ontologia.

Ao colega de pós-graduação Paulo Cordeiro, MsC, pelo exemplo de metodologia científica.

Ao Prof. Hugo Vieira Neto, PhD, pelo conhecimento do verdadeiro espírito científico.

A Rosângela Sarmiento Gonçalves, Analista de Sistemas do TJPR, pela iniciação ao domínio da jurisprudência.

Aos amigos Jefferson Fonseca Moreira e Josué Andrade Gomes pela revisão do Abstract.

RESUMO

MOLINARI, Alberto H. Indexação de Acórdãos por meio de uma Ontologia Jurisprudencial Populada a partir de um Corpus Jurídico Real. 190 f. Dissertação – Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2011.

Visando a uma melhoria na qualidade e precisão dos serviços de pesquisa jurisprudencial dos tribunais de justiça, a presente dissertação propõe a identificação automática de termos, ou sentenças, relevantes em um corpus de documentos jurisprudenciais. Cada termo identificado deve ser relacionado a um conceito que represente o seu significado no domínio da jurisprudência, compondo assim um indivíduo deste domínio. Ao final, cada indivíduo deve ser armazenado em uma base de conhecimentos, possibilitando, assim, pesquisas semânticas sobre os documentos. Para alcançar tais objetivos, são propostos métodos para a extração de sentenças relevantes e para a construção de uma ontologia de aplicação para representação de acórdãos. Na sequência são propostos ainda métodos de navegação e pesquisa na ontologia. Na extração de sentenças são abordadas técnicas de Mineração de Textos, tais como Extração de Sentenças, Análise de Expressões Regulares, *Stemming*, *Stop-words* e Vocabulários Controlados. A construção da ontologia segue a metodologia OTKM, utilizando-se também de linguagens de representação de conhecimento, tais como DL, RDF e OWL. Para a navegação na ontologia é abordada a *framework* Jena. Para pesquisas na ontologia é abordada a linguagem de consultas SPARQL. Para validar os métodos aqui propostos, foram construídos uma ontologia de aplicação para o domínio de acórdãos, bem como um aplicativo para gestão do conhecimento dos acórdãos baseado na ontologia. O aplicativo inclui rotinas de extração de conhecimento de um corpus de acórdãos, de população da ontologia com os conhecimentos extraídos e de pesquisa semântica sobre a ontologia populada. A ontologia bem como o conhecimento extraído de 50 acórdãos foram submetidos à crítica por especialistas em jurisprudência. Ao final, a rotina de pesquisa semântica foi submetida a uma experimentação com a ontologia populada por 15 mil acórdãos, todos extraídos da base de jurisprudência real do Tribunal de Justiça do Estado do Paraná. Os resultados obtidos nos experimentos demonstraram que a abordagem foi satisfatória tanto na indexação dos documentos como na pesquisa semântica, demonstrando que a ontologia desenvolvida responde aos requisitos da aplicação.

Palavras-chave: Recuperação de Informações; Ontologia; Pesquisa Semântica; Jurisprudência; OTKM; OWL; SPARQL

ABSTRACT

MOLINARI, Alberto H. Judgments Indexing through an Jurisprudential Ontology Populated from a Real Legal Corpus. 190 f. Dissertação – Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2011.

Aiming at improving the quality and accuracy of jurisprudential search services of the courts of justice, this dissertation proposes the automatic identification of relevant terms, or sentences, in a corpus of jurisprudential documents. Each identified term should be related to a concept that represents its meaning in the domain of jurisprudence, thus instancing an individual of that domain. At the end, each individual must be stored in a knowledge base, thus enabling semantic queries over the documents. To achieve these objectives, methods were proposed for the extraction of relevant sentences and to build an application ontology for the representation of judgments. Furthermore, ontology navigation and search methods were proposed. For the extraction of sentences, several Text Mining techniques were used, such as Sentence Extraction, Analysis of Regular Expressions, Stemming, Stop-words and Controlled Vocabularies. The construction of the ontology followed the methodology OTKM and used some knowledge representation languages, such as DL, RDF and OWL. The Jena framework was applied for navigation in the ontology. SPARQL queries language was applied for search in the ontology. To validate the methods here proposed, an application ontology for the domain of judgments were built, as well as a knowledge management application to the judgments based on the ontology. The application includes routines for knowledge extraction from a corpus of judgments, population of the ontology with the extracted knowledge and then semantic search on the populated ontology. The ontology and the knowledge extracted from 50 judgments were submitted to criticism by experts in jurisprudence. At the end, the semantic search routine was experimented with the ontology populated by 15,000 judgments, all extracted from the actual jurisprudence base of the Court of Paraná, a state of Brazil. The results obtained in the experiments demonstrated that the approach was satisfactory in both the indexing of documents as well as in the semantic search, showing that the developed ontology meets the application requirements.

Keywords: Information Retrieval; Ontology; Semantical Search; Jurisprudence; OTKM; OWL; SPARQL

LISTA DE FIGURAS

FIGURA 1 – TAXONOMIA PRESENTE NO TESAURO DO CJF	47
FIGURA 2 – UM EXEMPLO DE REDE	50
FIGURA 3 – TRIPLA DE AFIRMAÇÃO RDF	58
FIGURA 4 – GRAFO RDF	59
FIGURA 5 – GRAFO RDF	64
FIGURA 6 – O METAPROCESSO DE CONHECIMENTO	68
FIGURA 7 – O PROCESSO DE CONHECIMENTO	72
FIGURA 8 – PRECISÃO E REVOCAÇÃO – PREMISSAS	73
FIGURA 9 – PRECISÃO E REVOCAÇÃO – FATOS	73
FIGURA 10 – PRECISÃO	74
FIGURA 11 – REVOCAÇÃO	74
FIGURA 12 – DIAGRAMA DOS MÉTODOS PROPOSTOS	77
FIGURA 13 – DESCRIÇÃO SEMI-FORMAL DA ONTOLOGIA	83
FIGURA 14 – REFINAMENTO DA ONTOLOGIA SEMI-FORMAL	84
FIGURA 15 – VISUALIZAÇÃO DO PROTÓTIPO DA ONTOLOGIA JURISTJPR	85
FIGURA 16 – SEGUNDO REFINAMENTO DA ONTOLOGIA SEMI-FORMAL	87
FIGURA 17 – TERCEIRO REFINAMENTO DA ONTOLOGIA SEMI-FORMAL	90
FIGURA 18 – TERCEIRO REFINAMENTO (CONT.)	91
FIGURA 19 – COMPONENTES DA APLICAÇÃO	102
FIGURA 20 – ESTRUTURA DA TABELA DE DOCUMENTOS DE ACÓRDÃOS	103
FIGURA 21 – JANELA DO MÓDULO DE VISUALIZAÇÃO DO JURTREEVIS	104
FIGURA 22 – JANELA DE EXTRAÇÃO/POPULAÇÃO DA ONTOLOGIA	105
FIGURA 23 – VISUALIZAÇÃO DOS INDIVÍDUOS POPULADOS NA ONTOLOGIA	127
FIGURA 24 – RELACIONAMENTO LEGISLAÇÃO/ACÓRDÃO NA TAXONOMIA .	128
FIGURA 25 – JURTREEVIS – JANELA DE PESQUISA	129
FIGURA 26 – JURTREEVIS – COMPONENTES ADICIONAIS DA JANELA DE PES- QUISA	132
FIGURA 27 – RESULTADOS DAS PESQUISAS NOS SÍTIOS COM MAIS FUNCIO- NALIDADES	164

LISTA DE TABELAS

TABELA 1	– TÉCNICAS UTILIZADAS NOS TRABALHOS CORRELATOS	34
TABELA 2	– ESTRUTURA DE CLASSES DE CARACTERES	40
TABELA 3	– CLASSES DE CARACTERES PREDEFINIDAS	41
TABELA 4	– META-CARACTERES QUANTIFICADORES	41
TABELA 5	– PLANILHA DE VIABILIDADE	78
TABELA 5	– PLANILHA DE VIABILIDADE	79
TABELA 6	– DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS DA ONTOLOGIA (ORSO)	82
TABELA 7	– DEFINIÇÃO DE TAGS PARA REPRESENTAÇÃO DE EXTRAÇÃO DE CONCEITOS	107
TABELA 8	– CARACTERÍSTICAS ABORDADAS NOS MÉTODOS DE EXTRAÇÃO	107
TABELA 8	– CARACTERÍSTICAS ABORDADAS NOS MÉTODOS DE EXTRAÇÃO	108
TABELA 8	– CARACTERÍSTICAS ABORDADAS NOS MÉTODOS DE EXTRAÇÃO	109
TABELA 8	– CARACTERÍSTICAS ABORDADAS NOS MÉTODOS DE EXTRAÇÃO	110
TABELA 8	– CARACTERÍSTICAS ABORDADAS NOS MÉTODOS DE EXTRAÇÃO	111
TABELA 8	– CARACTERÍSTICAS ABORDADAS NOS MÉTODOS DE EXTRAÇÃO	112
TABELA 9	– EXEMPLO DE TAGS VARRIDAS PELO ALGORITMO DE POPULAÇÃO	124
TABELA 9	– EXEMPLO DE TAGS VARRIDAS PELO ALGORITMO DE POPULAÇÃO	125
TABELA 10	– RESULTADOS DAS CONSULTAS À LEI MARIA DA PENHA	140
TABELA 11	– RESULTADO DA AVALIAÇÃO COM ENFOQUE NO USUÁRIO	142
TABELA 11	– RESULTADO DA AVALIAÇÃO COM ENFOQUE NO USUÁRIO	143
TABELA 11	– RESULTADO DA AVALIAÇÃO COM ENFOQUE NO USUÁRIO	144
TABELA 12	– RESULTADO DA AVALIAÇÃO DA PESQUISA	147
TABELA 13	– FUNCIONALIDADES DE PESQUISA JURISPRUDENCIAL NOS TRIBUNAIS BRASILEIROS	158
TABELA 13	– FUNCIONALIDADES DE PESQUISA JURISPRUDENCIAL NOS TRIBUNAIS BRASILEIROS	159
TABELA 14	– PESQUISAS NOS SÍTIOS COM MAIS FUNCIONALIDADES	161
TABELA 15	– RESULTADO DA PESQUISA EM OUTROS TRIBUNAIS	164
TABELA 16	– URLS DAS PESQUISAS JURISPRUDENCIAIS CONSULTADAS NOS TRIBUNAIS	186
TABELA 16	– URLS DAS PESQUISAS JURISPRUDENCIAIS CONSULTADAS NOS TRIBUNAIS	187

LISTA DE SIGLAS

API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CEJ	Centro de Estudos Judiciários
CJF	Conselho da Justiça Federal
CNJ	Conselho Nacional de Justiça
CPGEI	Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial
DL	Lógica de Descrição
FOL	Lógica de Primeira-Ordem
HTML	HiperText Markup Language
KM	Knowledge Management
ORSD	Documento de Especificação de Requisitos da Ontologia
OTKM	On-To-Knowledge Methodology
OWL	OWL Web Ontology Language
PLN	Processamento de Linguagem Natural
RBC	Raciocínio Baseado em Casos
RDF	Resource Description Framework
RI	Recuperação de Informações
RTF	Rich Text Format
SGBD	Sistema Gerenciador de Base de Dados
SPARQL	Protocol And RDF Query Language
STF	Superior Tribunal Federal
STJ	Superior Tribunal de Justiça
TJAL	Tribunal de Justiça de Alagoas
TJAM	Tribunal de Justiça do Amazonas

TJAP	Tribunal de Justiça do Amapá
TJBA	Tribunal de Justiça da Bahia
TJCE	Tribunal de Justiça do Ceará
TJDFT	Tribunal de Justiça do Distrito Federal e dos Territórios
TJES	Tribunal de Justiça do Espírito Santo
TJGO	Tribunal de Justiça do Goiás
TJMA	Tribunal de Justiça do Maranhão
TJMG	Tribunal de Justiça de Minas Gerais
TJMS	Tribunal de Justiça do Mato Grosso do Sul
TJMT	Tribunal de Justiça do Mato Grosso
TJPA	Tribunal de Justiça do Pará
TJPB	Tribunal de Justiça da Paraíba
TJPE	Tribunal de Justiça de Pernambuco
TJPI	Tribunal de Justiça do Piauí
TJPR	Tribunal de Justiça do Estado do Paraná
TJRJ	Tribunal de Justiça do Rio de Janeiro
TJRN	Tribunal de Justiça do Rio Grande do Norte
TJRO	Tribunal de Justiça de Rondônia
TJRR	Tribunal de Justiça de Roraima
TJRS	Tribunal de Justiça do Rio Grande do Sul
TJSC	Tribunal de Justiça de Santa Catarina
TJSE	Tribunal de Justiça de Sergipe
TJSP	Tribunal de Justiça de São Paulo
TJTO	Tribunal de Justiça de Tocantins
TREDF	Tribunal Regional Eleitoral do Distrito Federal
TRF	Tribunal Regional Federal
TSE	Tribunal Superior Eleitoral

URI	Uniform Resource Identifier
URL	Uniform Resource Locator – Localizador Padrão de Recursos
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

SUMÁRIO

1 INTRODUÇÃO	12
1.1 PROBLEMA	13
1.2 JUSTIFICATIVA	14
1.3 OBJETIVOS	14
1.4 CONTRIBUIÇÕES CIENTÍFICAS	15
1.5 ORGANIZAÇÃO DOS CAPÍTULOS	16
2 REVISÃO BIBLIOGRÁFICA	18
2.1 DOCUMENTOS JURISPRUDENCIAIS	19
2.1.1 Documento de Acórdão	20
2.2 TRABALHOS CORRELATOS	21
2.2.1 Assumpção e Ribeiro Neto	21
2.2.2 Braga Júnior	22
2.2.3 Oliveira	23
2.2.4 Falcão	23
2.2.5 Farzindar e Lapalme	24
2.2.6 Maçoli	26
2.2.7 Alves	26
2.2.8 Cerqueira	28
2.2.9 Projeto ESTRELLA	28
2.2.10 Câmara Júnior	31
2.2.11 Zahra	32
2.2.12 Ramos Júnior e Rover	32
2.2.13 Conclusão	33
2.3 MINERAÇÃO DE TEXTOS	35
2.3.1 Recuperação de Informações	35
2.3.2 Extração de Sentenças	36
2.3.3 Análise de Expressões Regulares	38
2.3.4 <i>Stemming</i>	42
2.3.5 <i>Stop-Words</i>	43
2.3.6 Tesouros	44
2.4 GESTÃO DE CONHECIMENTO BASEADO EM ONTOLOGIAS	47
2.4.1 Representação de Conhecimento	48
2.4.1.1 Estruturas de Representação de Conhecimento Baseado em Rede e em DL	50
2.4.1.2 Aplicação de Conhecimento Representado em DL	53
2.4.2 Ontologias	56
2.4.3 Ontologias e a Web Semântica	57
2.4.3.1 Representação de Informações na Web	57
2.4.3.2 Linguagens de Representação de Ontologias na Web	60
2.4.3.3 Linguagem de Consultas em Ontologias na Web	63
2.4.3.4 Desenvolvimento de Aplicações Web Baseadas em Ontologia	65
2.4.4 Metodologias para desenvolvimento e gestão de ontologias	66

2.4.5 On-To-Knowledge Methodology (OTKM)	67
2.4.5.1 Metaprocesso de Conhecimento	67
2.4.5.2 Fase Estudo de Viabilidade	67
2.4.5.3 Fase Ponta-pé Inicial	69
2.4.5.4 Fase Refinamento	69
2.4.5.5 Fase Avaliação	70
2.4.5.6 Fase Aplicação e Evolução	71
2.4.5.7 Processo de Conhecimento	72
2.5 AVALIAÇÃO DE RESULTADOS DE PESQUISAS	73
2.5.1 Precisão	73
2.5.2 Revocação	74
3 MÉTODOS	76
3.1 METAPROCESSO DE CONHECIMENTO	78
3.1.1 Aplicação da Fase Estudo de Viabilidade	78
3.1.2 Aplicação da Fase Ponta-pé inicial	82
3.1.3 Aplicação da Fase Refinamento	83
3.1.3.1 Refinamento da ontologia semi-formal	83
3.1.3.2 Formalização da ontologia almejada e criação de um protótipo	85
3.1.4 Aplicação da Fase Avaliação	86
3.1.4.1 Avaliação com enfoque na ontologia	86
3.1.4.2 Avaliação com enfoque tecnológico	88
3.1.4.3 Avaliação com enfoque no usuário	88
3.1.5 Aplicação da Fase Aplicação e Evolução	88
3.2 CONSTRUÇÃO DA APLICAÇÃO DE GESTÃO DA ONTOLOGIA	100
3.2.1 Módulo de Extração de Corpus	103
3.2.2 Módulo de Visualização	103
3.2.3 Módulo de Extração	105
3.2.4 Módulo de População	119
3.2.5 Módulo de Pesquisa	128
3.3 AVALIAÇÃO DA ONTOLOGIA COM ENFOQUE NO USUÁRIO	136
3.3.1 Método da avaliação	136
3.4 AVALIAÇÃO DA PESQUISA SEMÂNTICA	137
3.4.1 Método da avaliação	138
4 RESULTADOS	142
4.1 AVALIAÇÃO DA EXTRAÇÃO DE INDIVÍDUOS	142
4.2 AVALIAÇÃO DA PESQUISA SEMÂNTICA	146
4.3 DISCUSSÃO DOS RESULTADOS	148
5 CONCLUSÃO	150
6 SUGESTÕES PARA TRABALHOS FUTUROS	152
REFERÊNCIAS	154
APÊNDICE A – COMPARATIVO ENTRE AS PESQUISAS JURISPRUDENCIAIS	158
APÊNDICE B – RESULTADO DO MÓDULO DE EXTRAÇÃO – EXEMPLO RE-	
SULTANTE DO PROCESSAMENTO DE UM ACÓRDÃO	166
APÊNDICE C – RESULTADO DO MÓDULO DE POPULAÇÃO – EXEMPLO RE-	
SULTANTE DO PROCESSAMENTO DE UM ACÓRDÃO	180
ANEXO A – EXEMPLO DE ACÓRDÃO	182
ANEXO B – URLS DAS PESQUISAS JURISPRUDENCIAIS DOS TRIBUNAIS ...	186

1 INTRODUÇÃO

Jurisprudência é uma palavra, na sua essência, resultante da conjunção dos termos latinos *jus*, que significa Direito, em português, e *prudentia*, que significa sabedoria, segundo Silva, Filho e Carvalho (2009). Assim jurisprudência pode ser compreendida como a aplicação do Direito com sabedoria.

Segundo Santos (2001), Jurisprudência é o conjunto das soluções uniformes dadas pelos tribunais às questões de Direito ou a autoridade dos casos julgados sucessivamente do mesmo modo. Assim, a Jurisprudência não se forma por decisões isoladas (o que consistiria em mero precedente), mas após uma série de decisões judiciais de mesmo entendimento.

A Jurisprudência é considerada fonte não formal do direito, ou seja não se trata de uma norma de aplicação obrigatória, ficando o magistrado livre para decidir conforme o seu próprio julgamento. Logo, um documento jurisprudencial não poderá, isoladamente, motivar uma sentença ou decisão judicial. Poderá, entretanto, embasar ou reforçar a conclusão do julgador. Uma decisão judicial, integrante da Jurisprudência, poderá ser utilizada formalmente como norma jurídica somente quando for transformada, pelo Superior Tribunal Federal (STF) em súmula vinculante.

Na prática, sempre que uma decisão judicial é publicada, passa a integrar o acervo jurisprudencial do tribunal onde se originou, independentemente da existência prévia de decisões correlatas. A validade da utilização de uma decisão no embasamento de documentos judiciais tais como petições, elaborados por advogados, bem como novas sentenças ou decisões, prolatadas por magistrados, depende exclusivamente do julgamento do redator, posto que ela pode consistir em um precedente ou pode compartilhar do direcionamento de outras decisões já publicadas, no mesmo ou em diversos tribunais. A consulta à Jurisprudência, quando da elaboração dos citados documentos, é muito comum, sendo recorrentes as referências a várias decisões ou citações existentes nas jurisprudências de todos os tribunais.

Após centenas de anos de Direito no Brasil, a jurisprudência de cada tribunal é extensa, sendo composta por milhares (e às vezes milhões, nos maiores tribunais) de documentos. Vi-

sando a tornar disponível a sua base jurisprudencial, os tribunais costumam oferecer ferramentas de busca nos seus sítios da Internet, baseadas, geralmente, em pesquisas textuais ou por metadados.

1.1 PROBLEMA

As consultas jurisprudenciais disponíveis, costumam possuir, na grande maioria dos tribunais, mecanismos de busca textual os quais não dispõem de funcionalidades minimamente desejáveis em sistemas deste tipo, tais como cálculo de relevância de cada documento resultante ou processamento linguístico de consultas e textos. Localizar o que se necessita, nos resultados das pesquisas, pode consistir em uma tarefa bem pouco produtiva, considerando que, muitas vezes, o retorno da pesquisa pode incluir várias dezenas, ou até centenas, de referências a decisões judiciais publicadas.

Para conhecer as soluções de consulta jurisprudencial já adotadas, as suas deficiências e as boas práticas, foram experimentados os sistemas da maioria dos tribunais de justiça estadual do país bem como de tribunais superiores, cujas URLs de acesso encontram-se listadas no sítio do Conselho Nacional de Justiça (CNJ), conforme o anexo B. Através de várias pesquisas, pôde-se efetuar uma comparação entre as buscas disponíveis em cada sítio, conforme o apêndice A.

Mais do que um levantamento de recursos disponíveis, a varredura nas pesquisas de todos os tribunais foi importante para observar o estado dos sistemas no setor, de onde se obtiveram as seguintes conclusões:

- pesquisas livres, por texto, costumam trazer resultados extensos e heterogêneos;
- pesquisas restritas costumam ser precisas, trazendo resultados reduzidos, mais pertinentes e homogêneos;

Ciente destas conclusões, é questionável o porque de a maioria dos tribunais não dispor de pesquisas restritas, questão esta que foi dirimida pela Seção de Jurisprudência do Tribunal de Justiça do Estado do Paraná (TJPR), através da constatação dos seguintes fatos:

- as pesquisas restritas dos tribunais são, geralmente, baseadas na indexação manual prévia dos documentos;
- a indexação de um documento jurídico é uma tarefa custosa que depende de ações de especialistas;

- as seções de jurisprudência dos tribunais estaduais geralmente dispõem de poucos especialistas, compondo um quadro insuficiente para efetuar a indexação de cada documento publicado.

1.2 JUSTIFICATIVA

A experimentação das buscas jurisprudenciais disponíveis demonstrou que uma pesquisa restrita eficaz provê ganhos de qualidade e, conseqüentemente, produtividade. A área de Recuperação de Informações (RI) tem, entre os seus objetivos, o aprimoramento das pesquisas textuais através de adições, às consultas originais, de hiperônimos, sinônimos e palavras coocorrentes, o que, segundo Park, Byrd e Boguraev (2003), geralmente amplia a eficiência das buscas, mas sofre de ambigüidades e dificuldades de referenciamento a recursos léxicos. Os usuários necessitam de funcionalidades mais inteligentes e versáteis. Segundo Buttler et al. (2002), isso pode ser alcançado através de motores de busca compatíveis com a Web Semântica ou baseadas em ontologias, as quais, segundo Sheth, Arpinar e Kashyap (2003), proveem descrições de conceitos de domínios e seus inter-relacionamentos.

A Web Semântica é uma extensão da atual web, onde as informações são dotadas de significados bem definidos, possibilitando que computadores e pessoas trabalhem cooperativamente (BERNERS-LEE; HENDLER; LASSILA, 2001). Assim, as atividades na Web Semântica objetivam adicionar à teia de hipertextos existente uma camada semântica compreensível por computador incluindo a criação de anotações semânticas bem como de conexão entre páginas web e ontologias (PARK; BYRD; BOGURAEV, 2003).

1.3 OBJETIVOS

O objetivo principal deste trabalho é **suprir a demanda de indexação de documentos jurisprudenciais, do tipo acórdão, de forma automática e que obtenha vantagens de uma representação de conhecimentos baseada em ontologias visando a disponibilizar uma pesquisa semântica de jurisprudência que traga ganhos de precisão e revocação nos resultados**. Decompondo o principal objetivo, tem-se como objetivos secundários:

Indexação Automática: pretende-se gerar automaticamente os índices desejáveis para cada acórdão¹ do TJPR, através da utilização de técnicas de RI. A escolha deste tipo de docu-

¹ Acórdão é a manifestação de um órgão judicial colegiado, que externa um posicionamento argumentado sobre a aplicabilidade de determinado direito a uma situação fática específica (TSE, 2010).

mento jurisprudencial e deste tribunal como fonte de informações, se deve aos seguintes fatores:

- autorização de acesso que o autor obteve à base de dados jurisprudencial de testes do TJPR (a qual consiste em uma cópia da base de produção);
- necessidade de se restringir o campo de pesquisa, viabilizando assim a realização do presente trabalho em tempo limitado;

Conceitualização: cada índice gerado deve ser conceitualizado, ou seja, deve ser identificado e relacionado a um conceito que represente o seu significado no domínio da jurisprudência;

Base de Conhecimento: os índices extraídos de cada acórdão devem ser armazenados em uma ontologia que seja expressiva o suficiente para representar o domínio da jurisprudência do TJPR, sendo que cada índice deverá ser distinguido pelo conceito que representa no domínio;

Pesquisa Semântica: a ontologia, populada pelos índices de cada acórdão, deverá ser submetida a uma pesquisa que faça distinção entre os tipos de conceitos a serem pesquisados, de forma que o usuário possa, além de informar os termos da pesquisa, selecionar o tipo de conceito onde os termos deverão ser buscados.

1.4 CONTRIBUIÇÕES CIENTÍFICAS

O grupo de pesquisas MEMENTO do Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial (CPGEI), há sete anos dedica-se a realizar pesquisas nos temas de metodologias de aquisição de conhecimentos para o desenvolvimento de ontologias, alinhamento de ontologias, bem como de técnicas de desenvolvimento de Web Semântica.

Neste contexto, o presente trabalho pretende trazer uma contribuição ao citado grupo: a realização do ciclo completo de uma aplicação baseada em ontologia, o qual se inicia pelo desenvolvimento da ontologia, passando pela sua validação por especialistas de domínio e chegando à criação de um aplicativo que visa à utilização da ontologia. Além disso, pretende-se neste trabalho oferecer as seguintes contribuições à comunidade científica:

experimentação com massa de dados reais, pouco encontrada na literatura, principalmente do Brasil, e mesmo quando existente, muitas vezes dispõe de pequena quantidade de casos. Exemplos: (MAÇOLI, 2005), (OLIVEIRA, 2008);

criação de uma ontologia de aplicação para descrever acórdãos,² visto que as ontologias já desenvolvidas são de alto nível³ ou do domínio do Direito, sendo muito genéricas ao descrever o vocabulário relacionado e tornando-se, assim, distantes das aplicações existentes no ramo do Direito, inclusive a Jurisprudência. Exemplos: (BREUKER et al., 2007), (ALVES, 2005), (CERQUEIRA, 2007);

conceitualização de documentos jurídicos, visto que muitos trabalhos limitam-se a indexar os documentos, ou processá-los atômicamente, não relacionando os índices a qualquer conceito que poderia os definir. Exemplos: (ASSUMPCÃO; NETO, 2001), (OLIVEIRA, 2008), (FARZINDAR; LAPALME, 2004), (JÚNIOR, 2007);

conceitualização automática de documentos, visto que os métodos já existentes às vezes tem dependência de interação com o usuário. Exemplos: (FALCÃO, 2003), (JÚNIOR, 2007), (ZAHRA, 2009) ;

conceitualização independente de treinamento, visto que dentre os métodos existentes alguns dependem de bases de dados com casos treinados ou levantados a priori. Exemplos: (JÚNIOR, 2007), (OLIVEIRA, 2008);

entendimento e detalhamento da metodologia OTKM, visto que a literatura a respeito (YORK; STEFFEN; STUDER, 2003) é superficial.

1.5 ORGANIZAÇÃO DOS CAPÍTULOS

O capítulo 2 apresenta uma revisão dos assuntos que afetam ou embasam os métodos utilizados no presente trabalho. Inicialmente são descritos os documentos envolvidos na jurisprudência, sendo em seguida detalhado o documento do tipo Acórdão. Na sequência são apresentados resumos analíticos de doze trabalhos, entre artigos e dissertações, já publicados à respeito dos temas do Direito, da Jurisprudência e da extração e gestão do conhecimento. Em seguida são apresentados vários tópicos de Mineração de Textos, visando à extração de conhecimento de um *corpus*, e de Gestão de Conhecimento, que visam ao processamento dos conhecimentos extraídos. Por fim é abordada uma técnica para avaliação de resultados de pesquisas.

²**Ontologias de aplicação** descrevem conceitos relativos a um determinado domínio e a uma tarefa específica. Tais conceitos muitas vezes correspondem a papéis desempenhados por entidades de domínio ao executar uma determinada atividade (GUARINO, 1998).

³**Ontologias de Alto Nível** descrevem conceitos muito gerais como espaço, tempo, matéria, objeto, evento, ação, etc, que são independentes de um problema particular ou domínio (GUARINO, 1998).

O capítulo 3 descreve os métodos empregados na criação da ontologia que representa o domínio dos acórdãos da jurisprudência do TJPR, bem como da respectiva construção da aplicação de gestão. Ao longo do capítulo são apresentados artefatos intermediários correspondentes a cada passo dos métodos. Ao final são descritos os critérios utilizados para a avaliação dos resultados do trabalho, os quais são apresentados e analisados no capítulo 4.

No capítulo 5 são apresentadas conclusões e contribuições às quais o trabalho conduziu.

No capítulo 6 são efetuadas sugestões para estudos ou trabalhos futuros.

No apêndice A é efetuada uma comparação entre as pesquisas jurisprudências disponíveis nos sites da maioria dos tribunais de justiça estaduais (incluindo tribunais superiores), cujas URLs são listadas no anexo B.

No apêndice B é apresentado, por completo, um artefato intermediário dos métodos: o conhecimento extraído do acórdão apresentado (parcialmente) no anexo A, o qual foi extraído da base de dados do TJPR.

No apêndice C é apresentado mais um artefato intermediário: o conhecimento de um acórdão que populou a ontologia, em formato de relatório visando à leitura por especialistas.

2 REVISÃO BIBLIOGRÁFICA

Para embasar os objetivos deste trabalho efetuou-se uma revisão bibliográfica dividida em quatro frentes de estudo:

Documentos Jurisprudenciais – visa a entender quais são, e como são, os documentos publicados na jurisprudência brasileira;

Trabalhos Correlatos – antes de selecionar técnicas ou elaborar hipóteses de solução para o presente trabalho é importante consultar e se posicionar a respeito do que já foi estudado, proposto ou criado em torno da problemática em questão;

Indexação de Documentos – é necessário pesquisar técnicas que auxiliem na indexação que vai classificar ou definir cada documento do tipo acórdão do TJPR. Naturalmente que tais técnicas presumem conhecimentos ou experiências prévias no domínio em questão, posto que não existe tecnologia capaz de produzir conhecimento sem informações iniciais. Colocados estes requisitos, fica claro que a pesquisa deve se dar dentro do arcabouço de técnicas da Inteligência Artificial, mais especificamente no ramo da Mineração de Textos, visto que:

“Mineração de Textos” vem sendo crescentemente utilizada para denotar todas as tarefas as quais, através da análise de grandes quantidades de texto e detectando o uso de padrões, tentam extrair prováveis informações úteis (desde que provavelmente corretas) (SEBASTIANI, 2002).

Desenvolvimento de ontologias e bases de conhecimento – na sequência é necessária a pesquisa de metodologias adequadas para sistematizar o conhecimento adquirido em cada documento através de uma ontologia.

2.1 DOCUMENTOS JURISPRUDENCIAIS

Visto que a jurisprudência consiste na interpretação das leis, segundo magistrados, as quais estão disponíveis através do acervo de documentos judiciais decisórios, deve-se entender quais, e como, são tais documentos:

Sentença: ¹ trata-se do documento resultante da prolação de uma sentença (de condenação ou absolvição) por um juiz de direito, como resultado de um processo em primeira instância da justiça. Quando uma das partes envolvidas no processo, discorda da decisão do juiz, tem o direito de recorrer da sentença, através do envio do processo para a segunda instância da justiça, onde a decisão do juiz será avaliada e julgada por um ou mais desembargadores, visando ao seu endosso ou reforma através de um dos dois documentos a seguir;

Decisão Monocrática: quando o recurso à sentença refere-se a uma questão que já foi claramente julgada em decisões passadas em segunda instância ou em súmulas², o relator do recurso pode prolar uma Decisão Monocrática, na qual ele segue sozinho, e de maneira sumária, as decisões passadas.

Acórdão: ³ quando o tema do recurso não encontra similaridade em decisões passadas ou súmulas existentes, necessita de julgamento através de um órgão julgador, composto de um relator e pelo menos mais dois magistrados. O relator formula um documento, o qual é defendido em uma sessão de julgamento, composto por um relatório do teor do processo originário e do conteúdo do recurso proposto, da fundamentação do julgamento do relator seguido da sua decisão a qual é votada pelos demais componentes, resultando, ao final do documento, no voto da maioria.

Os três documentos citados, além de se apresentarem na ordem de ocorrência no sistema judiciário, também se encontram em ordem de influência na jurisprudência, logo no embasamento jurídico, significando que o acórdão é o documento de maior relevância no estudo jurisprudencial.

¹ **Sentença:** decisão que resolve a causa ou questão controvertida sobre a relação de direito litigioso. Em juízo criminal é a legítima decisão da causa feita por juiz competente, segundo a lei, e as decisões do júri, e a prova dos autos (SANTOS, 2001).

² **Súmula:** coleção de três acórdãos, no mínimo, de um mesmo tribunal, nos quais se adota a mesma exposição de preceito jurídico em tese (SANTOS, 2001). Decisões similares em segunda instância a respeito de um determinado tema, transformadas em lei após repetidas reincidências, segundo julgamento em terceira instância.

³ **Acórdão:** de acordam, ou seja, concordam; decisão proferida em grau de recurso por tribunal coletivo e superior (SANTOS, 2001).

2.1.1 Documento de Acórdão

Sendo o acórdão o documento de maior relevância para a jurisprudência, também trata-se daquele onde há maior estruturação e padronização. Não há regulamentação do formato dos acórdãos a serem prolatados pelos tribunais brasileiros, entretanto há um padrão tácito, quase sempre seguido por todos os magistrados, resultante, provavelmente, de senso comum.

Cabeçalho: consiste nas primeiras linhas do documento e visa a introduzir metadados do processo originário, sendo que na primeira linha geralmente se insere tipo e número do recurso além do juizado de onde se originou o processo e nas linhas seguintes tipos e nomes de partes envolvidas;

Ementa: após algumas linhas em branco segue-se um parágrafo constituído por uma sequência de sentenças separadas por traço ou ponto. Esta seção é resultante do resumo que o relator elabora após a conclusão do relatório e visa a condensar em um parágrafo informações fáticas e legais que melhor definem o conteúdo do documento;

Relatório: após mais algumas linhas em branco inicia-se o relatório, o qual, geralmente, contém um resumo do processo original; os argumentos que fundamentaram o recurso; uma fundamentação que embasa o pensamento do relator a respeito do processo e do recurso (geralmente permeada de citações a leis, outros acórdãos, pensadores, fatos, etc.); e, ao final, o “voto” do relator, ou seja, a decisão que o relator propõe ao órgão julgador.

Decisão: no último parágrafo do texto consta a apreciação e o voto que os integrantes do órgão julgador fizeram sobre o relatório apresentado pelo relator, concluindo assim com a decisão que a maioria dos magistrados participantes **acordaram** a respeito do recurso (daí o nome **Acórdão**), os nomes dos votantes e a data e o local;

Assinatura: ao final consta o nome do relator e a sua assinatura.

No anexo A há um exemplar de um acórdão, extraído da base de dados do sistema jurisprudencial do TJPR, o qual foi recortado visando a resumir em poucas páginas a estrutura geralmente apresentada por um documento deste tipo e foi adicionado de anotações sobre os locais e os termos encontrados nos documentos dos quais se podem extrair sintaxes e semânticas.

2.2 TRABALHOS CORRELATOS

Visando à investigação da produção intelectual, foram revisados trabalhos científicos, relacionados ao contexto desta dissertação, onde se abordou a modelagem do domínio do Direito, a extração de conhecimento e a classificação automática de documentos jurídicos, visando a embasar, ou servir de ponto de partida para a solução do problema aqui proposto. Pretende-se, assim, apresentar um resumo dos trabalhos, bem como de seus pontos positivos e negativos **sob a ótica dos objetivos traçados para o presente trabalho**, sendo que, para se obter mais detalhes dos mesmos e das respectivas técnicas utilizadas, pode-se consultar a íntegra do texto de cada trabalho.

Segundo Cerqueira (2007), apesar de a Ontologia ter se desenvolvido principalmente nos Estados Unidos, foi na Europa que a modelagem dos domínios do Direito foi mais difundida, principalmente na Universidade de Amsterdã, devido, entre outros fatores, à necessidade de integração de sistemas legais nacionais na União Europeia, refletida na manutenção de inúmeros projetos na área de governo eletrônico e tecnologia da informação jurídica. No Brasil, embora a pesquisa em informática no ramo jurídico seja extensa, o tópico de conceitualizações ainda é incipiente. Por isso, esta revisão almeja verificar a que ponto chegaram, até o presente momento, os trabalhos de modelagem de domínios legais, preferencialmente os especializados no domínio da jurisprudência, bem como de aplicações de classificação ou indexação de documentos, jurídicos ou não, tanto na comunidade nacional quanto internacional. Os resumos são introduzidos no texto por ordem cronológica ou por afinidade de abordagem.

2.2.1 Assumpção e Ribeiro Neto

O trabalho de Assumpção e Neto (2001) objetiva obter informações semânticas através do tesouro (ver subseção 2.3.6, página 44) do Conselho da Justiça Federal (CJF) e da estrutura de documentos, visando a propor um modelo de recuperação de documentos da jurisprudência do STJ e do TJMG que leve a consultas com maior precisão no domínio do Direito. Para tal, o método propõe a expansão da consulta através de cálculo de similaridade das consultas e dos documentos com os descritores do tesouro através de Redes Bayesianas de Crenças. Leva-se, ainda, em consideração localizações nos documentos – ementa, indexação e decisão – visando à experimentação em quatro cenários, três baseadas nas localizações especificadas e um no documento todo. As experimentações demonstraram aumento da precisão dos resultados quando comparados a um modelo vetorial tradicional de pesquisa, por agregar evidências relativas ao domínio em questão.

Características favoráveis ao presente trabalho: apresentou o tesouro do CJF como fonte de informação semântica do domínio do Direito e demonstrou que abordagens de pesquisa que se utilizem de bases de informação específicas previamente elaboradas para o domínio em questão se constituem de importante fonte para o cálculo de relevância de termos e documentos.

Características desfavoráveis ao presente trabalho: o método proposto não gera classificação ou conceitualização dos documentos, característica esta desejável sob o enfoque do presente trabalho.

2.2.2 Braga Júnior

Júnior (2001) busca uma ampliação da precisão do resultado das pesquisas jurisprudenciais do Tribunal Regional Federal (TRF) através do Raciocínio Baseado em Casos (RBC), uma técnica que seleciona casos anteriores e semelhantes, em concordância com as informações de uma situação específica, a qual, neste trabalho, visa à identificação de acórdãos por similaridade. O RBC parte do conhecimento representado através de *scripts*, tríades objeto-atributo-valor, regras, redes semânticas, frames, formulários ou índices, os quais representam os casos anteriores, adquiridos previamente para serem utilizados em um raciocínio de analogia.

Em Júnior (2001) o conhecimento anterior é baseado em índices extraídos de metadados oriundos da base de dados originária dos acórdãos (número do processo, grupo, juiz, órgão julgador, datas de publicação e decisão) e do texto da ementa de cada documento. A consulta é expandida através da busca dos termos em um tesouro jurídico visando a agregar termos associados ao original, formando assim um caso de entrada. A partir do caso de entrada montado, aplica-se uma métrica de similaridade (através do algoritmo *Vizinho-mais-próximo*) para buscar casos semelhantes no banco de dados. Os casos encontrados são classificados segundo o grau de similaridade e exibidos para consulta.

Características favoráveis ao presente trabalho: analogamente ao trabalho anterior, demonstrou as vantagens do uso de uma base de conhecimento prévia, através, também neste caso, de um tesouro jurisprudencial, visando a conhecimentos semânticos a priori. Além disso, aponta que a modelagem prévia do conhecimento do domínio em questão pode ser favorável a uma abordagem de inteligência artificial.

Características desfavoráveis ao presente trabalho: não apresentou experimentação, validação ou resultados do método proposto, donde não se pode chegar a conclusões sobre a sua viabilidade para a solução do problema exposto.

2.2.3 Oliveira

Oliveira (2008) parte da dissertação de Júnior (2001) para propor um modelo de recuperação de informações baseado em casos oriundos do sistema de jurisprudência do Tribunal Regional Eleitoral do Distrito Federal (TREDF), visando à verificação do seu índice de precisão no resultado de busca por informações jurisprudenciais. O objetivo principal do trabalho é responder à uma questão que parece não ter sido dirimida em Júnior (2001): “a RBC, aplicada a um modelo de recuperação inteligente de jurisprudência, produzirá melhores índices de precisão, se comparados ao modelo tradicional vigente no TREDF?”

Os casos anteriores foram construídos através de *tópicos*, especificações das necessidades de informação dos especialistas voluntários, formuladas em linguagem natural, totalizando 25 tópicos. Estes tópicos foram convertidos em consultas, as quais foram submetidas ao sistema de pesquisa atual do tribunal, de cujo resultado os especialistas selecionaram os 10 documentos mais relevantes em relação ao tópico, na opinião deles.

A expansão de consulta, busca por similaridade e *ranking* correram de forma semelhante a Júnior (2001), porém utilizando o tesouro do Tribunal Superior Eleitoral (TSE). Cada um dos 25 tópicos foi submetido ao protótipo produzido pelo autor, sendo a precisão dos resultados das pesquisas baseada nos 10 documentos mais relevantes para cada tópico, conforme previamente selecionados pelos especialistas. Segundo a metodologia de avaliação adotada, os resultados alcançaram os objetivos do trabalho, por alcançarem índices maiores do que a pesquisa tradicional.

Características favoráveis ao presente trabalho: uso vantajoso de tesouro jurídico para ampliação de consulta.

Características desfavoráveis ao presente trabalho: a busca por similaridade se dá pelos documentos inteiros, não levando em consideração conceitos isolados que poderiam levar a diversas classificações para cada documento.

2.2.4 Falcão

Falcão (2003) almeja a proposição, aplicação e avaliação de um método de Classificação Temática em bases de dados textuais indexadas com uso de Vocabulário Controlado, visando a auxiliar na contextualização e atribuição de significado e propósito aos textos armazenados, possibilitando assim a descoberta de conhecimentos. Os textos são oriundos de discursos proferidos por deputados da Câmara dos Deputados Federal. Vocabulário controlado consiste em

uma lista de termos autorizados em determinado domínio, a qual não inclui relações de hierarquia e relação entre os termos (como nos tesauros). Cada discurso é previamente indexado por especialistas, baseado no vocabulário controlado, o qual inclui 607 descritores. Junto com especialistas foi definida uma lista de 14 temas. Cada descritor da lista é associado, por especialista, a um tema. Baseado nos seus índices, e na relação descritor-tema, cada documento é associado a um ou mais temas, obtendo assim a classificação temática de cada um.

Características favoráveis ao presente trabalho: extração de semântica a partir de vocabulário controlado previamente elaborado por especialistas. O método alcança o objetivo de classificação de documentos.

Características desfavoráveis ao presente trabalho: metodologia fortemente dependente da atuação manual de especialistas.

2.2.5 Farzindar e Lapalme

O artigo de Farzindar e Lapalme (2004) apresenta uma abordagem para resumir o registro legal de processos e decisões da corte federal do Canadá, as quais constituem a jurisprudência precedente da lei e apresentá-lo como um sumário em forma de tabela para as necessidades de advogados e especialistas no domínio jurídico.

O método apresentado investiga a extração das unidades mais importantes com base na identificação da estrutura temática do documento e da determinação dos temas argumentativos das unidades textuais do julgamento. Para superar as dificuldades impostas pela complexidade do domínio, da terminologia específica e das interpretações de expressões jurídicas, as quais produzem muitas ambiguidades, foi construído um dicionário conceitual contendo 200 conceitos de domínio jurídico.

Segmentos temáticos fornecem importantes informações que podem ser utilizados para responder questões específicas sobre o tema. Assim, este artigo descreve como lidar com o problema da exploração da estrutura do documento e seleção de conteúdo, de acordo com os temas de uma sentença.

Baseado em localização e sinalização faz o pré-processamento, onde identifica o corpo do texto, bem como a Segmentação Temática, onde o texto é dividido em 4 temas (Introdução, Contexto, Análise Jurídica e Conclusão) para a qual se utiliza também de certos marcadores linguísticos. Na *Filtragem* são eliminadas partes do texto de pouca relevância, como parágrafos ou sentenças de citações, exceto as referências à legislação existentes na Decisão. Depois são

filtrados dois tipos de segmentos: propostas e argumentos, os quais introduzem pontos de vista das partes e citações a tópicos anteriores ou a legislação (quando da eliminação de citações a leis, as referências são guardadas).

As citações são identificadas através de dois tipos de marcações: diretas e indiretas. Um marcador direto é um indicador linguístico, que se classifica em três categorias:

- **verbos** – por exemplo *concluir, definir, indicar, deferir, interpretar, recorrer, referir, proferir, declarar, resumir*;
- **conceitos** – substantivos, advérbios, adjetivos, por exemplo *seguinte, seção, subseção, página, parágrafo, nos termos*;
- **indicações complementares** – números, certas preposições, orações relativas e marcas tipográficas (dois pontos, aspas).

As citações indiretas são as unidades vizinhas de uma frase citada. Por exemplo, o segmento de citação na frase “*parágrafo 78(1), que diz o seguinte:*” é identificado por meio de marcadores diretos, mas ele aponta para as unidades textuais sem marcador direto, que são também citações. Assim, identificam-se as sentenças enumeradas na sequência de uma sentença citada para determinar um grupo de citações.

Na *Seleção* cria-se uma lista das melhores unidades candidatas para cada nível estrutural do resumo. O aplicativo calcula uma pontuação para cada sentença no julgamento com base em funções heurísticas relacionadas com as seguintes informações: a posição dos parágrafos no documento, a posição dos parágrafos no segmento temático, a posição das sentenças no parágrafo, a distribuição das palavras no documento e no corpus (tfidf). A *Produção* do resumo final controla o tamanho do resumo e exhibe as sentenças selecionadas em formato tabular. O resumo final é cerca de 10% do documento original.

Características favoráveis ao presente trabalho: reduz 90% a extensão do documento através de heurísticas baseadas em localização, sinalização e ocorrência que visam a selecionar o que é relevante no documento bem como agrupar as sentenças em quatro seções. o método independe de interação de usuários.

Características desfavoráveis ao presente trabalho: Reduz o texto, facilitando a interpretação do usuário final, mas não o classifica e nem o conceitualiza.

2.2.6 Maçoli

Maçoli (2005) propõe uma análise documentária de ementas, que consiste no estudo do documento, de sua integridade, componentes, separação de assunto, autor, enfim, uma análise do documento, para que, posteriormente, seja possível tratá-lo, armazená-lo e recuperá-lo com sucesso. Para que este processo seja executado é necessário utilizar métodos de análise e de tratamento dos documentos, tais como: decomposição; condensação de texto; representação através de linguagens documentárias, tais como vocabulários controlados e tesouros.

Acórdãos possuem elementos de dois tipos: descritivos – relacionados a dados contidos no texto – tais como nome do tribunal, turma que julgou o recurso, nome do juiz relator, partes, natureza, número do acórdão e etc; temáticos – ou de conteúdo – relacionados à aplicação do direito. O trabalho propõe:

- “buscar parâmetros metodológicos para a elaboração da Jurisprudência, oferecer alternativas na natureza operacional para a sua construção e desenvolver uma referência didática sobre sua elaboração”;
- apresentar e implementar uma metodologia de análise documentária aplicada à Jurisprudência Cível do TJSP, criando-se ementas capazes de identificar o principal conteúdo existente em cada uma delas, além da sugestão da criação de uma padronização na sua elaboração, tanto para efeito de armazenamento quanto de recuperação e tratamento. A metodologia consiste em indexação e representação temática de acórdãos em quatro categorias: instituto jurídico, fato, entendimento e argumento, visando a representar o raciocínio intrínseco ao acórdão.

Características favoráveis ao presente trabalho: o trabalho traz uma minuciosa conceituação dos elementos que compõem um acórdão a serem identificados e extraídos através da metodologia proposta;

Características desfavoráveis ao presente trabalho: a metodologia utilizada não foi apresentada, passando o trabalho diretamente para a experimentação, a qual consistiu no processamento de uma pequena amostra de dez acórdãos.

2.2.7 Alves

O trabalho de Alves (2005) se propõe a realizar um estudo descritivo de verbos do domínio jurídico com vistas à construção de uma ontologia que contribua para o aperfeiçoamento de

sistemas de Processamento de Linguagem Natural (PLN), especialmente para a construção de uma ferramenta de busca e extração de informações na Web.

A hipótese é que “informações de ordem sintático-semântica, baseadas em tipos de situação e papéis semânticos auxiliarão o sistema de pesquisa a interpretar e construir frases coerentemente. Inserir informações sobre a estrutura argumental de um verbo (*específico do domínio jurídico*) em um léxico computacional possibilitará que o sistema de busca reconheça que ele exige a presença de um conjunto X de argumentos e que estes precisam ser representados por conceitos de determinado tipo.”

Assim são selecionados verbos relevantes ao jargão jurídico, os quais são definidos através de uma Estrutura Ontológica Verbal, a qual, por sua vez, é sistematizada em quatro níveis:

Definição: informação para entendimento humano, mas incompreensível para o sistema, baseada em dicionário;

Relações lógico-semânticas: significado dos verbos no escopo da Semântica Lexical, visando a identificar as relações lógico-semânticas entre os verbos, baseado em relações expressas na *WordNet*;

Papéis semânticos significado dos verbos no nível sintático-semântico, visando à identificação dos papéis semânticos participantes e não-participantes do verbo, tais como agente e paciente;

Elementos *Frame*: descrições do papel contextual que cada complemento do verbo desempenha na situação evocada pelo verbo representadas através de *frames*.

Assim, na ontologia, construída para o trabalho, foram definidos 10 verbos – condenar, absolver, julgar, acordar, conceder, revogar, concluir, alegar, provar e recorrer – todos através de análise linguística manual. A criação computacional da ontologia se deu através do Protégé, ficando assim disponível para futuras aplicações de PLN que podem utilizar a ontologia como base de aprendizado inicial, visando ao reconhecimento de padrões linguísticos através dos quais se possa inferir sobre o conteúdo expresso em documentos jurídicos.

Características favoráveis ao presente trabalho: análise linguística extremamente detalhada visando à composição da ontologia. A ontologia possibilita a conceitualização de documentos jurídicos.

Características desfavoráveis ao presente trabalho: não favorece inferências baseadas em

temas recorrentes na jurisprudência tais como legislação e assunto, conceitos estes muito utilizados nas buscas jurisprudenciais.

2.2.8 Cerqueira

O trabalho de Cerqueira (2007) consiste em um “projeto interdisciplinar, onde se propõe um método para a construção de um modelo domínio-ontológico jurídico direcionado ao Direito Positivo brasileiro, sendo aplicado na modelagem do Imposto sobre Transmissão de Bens Imóveis por ato oneroso inter vivos (ITBI) do Município de Belo Horizonte, em linguagem OWL, com o uso do editor de ontologias Protégé.”

Para a modelagem da ontologia o autor propôs uma metodologia de desenvolvimento dividida em seis fases. “A primeira estabelece os critérios gerais, objetivos e domínio do modelo pretendido, sendo seguida pela fase de definição de sua estrutura formal de modelagem. A terceira fase investiga, sob o ponto de vista da inter-operabilidade, as ontologias jurídicas existentes, validando a adequação de seu reuso e explicitando o comprometimento ontológico inicial. A quarta fase demarca os metadados da ontologia, utilizados na documentação e na ligação da mesma, por questões de compatibilidade, com classificações distintas, tais como o tesouro jurídico do CJF e a CDD (Classificação Decimal do Direito). A quinta fase inicia a iteração propriamente dita de construção completa da ontologia de domínio, valendo-se concomitantemente de metodologia da Ciência do Direito. Por fim, testes de concordância entre os requisitos e a base de conhecimento são realizados na sexta fase, que encerra o método”. O tesouro jurídico do CJF foi importado para a ontologia, visando à conexão dos conceitos específicos aos descritores.

Características favoráveis ao presente trabalho: partindo de ontologias de alto nível, já produzidas em trabalhos anteriores, o trabalho estende a definição de conceitos específicos ao domínio em questão. A ontologia construída está disponível na Internet (<http://cerqueira.pro.br/owl/lop/1.0/lop.owl>)

Características desfavoráveis ao presente trabalho: o domínio modelado é muito distante do domínio da jurisprudência.

2.2.9 Projeto ESTRELLA

Breuker et al. (2007) descreve a metodologia utilizada para desenvolver a ontologia LKIF-Core. O plano de trabalho sugere que, a partir de uma ontologia de alto nível já existente – LRI-Core – o processo de desenvolvimento poderia ser uma abordagem *top-down* bastante simples.

Mas este não foi o caso, pois o número de conceitos jurídicos em LRI-Core era muito pequeno, tratando-se de uma ontologia de alto nível que cobre conceitos abstratos de senso-comum ao invés do campo do Direito.

Foram coletados entre os possíveis beneficiários deste projeto – cidadãos, profissionais legalistas e estudantes de profissões legalistas – termos jurídicos básicos. A lista integrou cerca de 250 termos, os quais foram pontuados, por parceiros do projeto Estrella, em sua relevância para o Direito, no seu nível de abstração e em sua especificidade jurídica, visando a definir um conjunto básico de cerca de 50 termos. LKIF-Core é atualmente a ontologia (de alto nível) mais abrangente e bem estruturada para o Direito.

Um número de grupos de conceitos centrais e relativamente abstratos foram identificados e modelados mais ou menos independentemente como módulos de uma ontologia. Um grupo é um conjunto de termos que têm um alto grau de (mútuas) relações podendo ser conceituados em relativo isolamento. Exemplos são as noções de 'norma' e 'ação'. Foram identificados dez grupos básicos, que cobrem mais ou menos a lista dos principais termos jurídicos. Ao final a quantidade de grupos totalizou dezessete, que são:

Top (topo): a LKIF é amplamente baseada no alto nível da LRI-Core, mas tem menos compromisso ontológico, no sentido de que impõe menos restrições sobre subclasses das categorias de topo;

Expression (expressão): abrange um número de primitivas de representação necessários para descrever estados mentais relacionais que ligam uma pessoa a uma proposição: por exemplo, crenças, declarações, etc.;

Norm (norma): define os conceitos mais centrais para a LKIF: por exemplo, norma, obrigação, proibição, etc.;

Processes (processos): descreve os conceitos relacionados com mudanças (involuntárias);

Action (ação): cobre os conceitos relacionados a ações e sua relação com mudanças físicas e intenções, por exemplo, ação, agente, etc.

Role (papéis): descreve construções subjacentes aos papéis a ser desempenhados pelos agentes;

Legal-action (ação jurídica): este módulo estende o módulo de ação com uma série de conceitos jurídicos relacionados a ações e agentes, tais como atos públicos, órgãos públicos, pessoa jurídica, pessoa física, etc.;

Legal-role (papel-jurídico): este módulo estende o módulo de papel com um pequeno número de conceitos jurídicos relacionados a papéis, profissões jurídicas, etc;

Place (Local): define primitivas de representação para descrever lugares, localizações e as relações entre eles;

Time (tempo): cobre primitivas de representação para descrever intervalos de tempo;

Spacetime (espaço-tempo): este módulo é um espaço reservado para os módulos de local e tempo;

Mereology (mereologia): descreve classes e propriedades que permitem expressar relações mereológicas, por exemplo, parcialidade, componentes, etc.

Relational Role (papel relacional): argumentos de uma relação, ou seja, o domínio e o intervalo de uma propriedade em OWL. Exemplos prototípicos de papéis relacionais geralmente são tomadas a partir do domínio família: pai, mãe, pai, filho;

Participant Role (papel participante): O papel que uma entidade executa em relação a um determinado processo em curso. Exemplos disso são o orador e o ouvinte em um processo de conversação;

Social Role (papel social) papéis sociais que apresentam propriedades individuais: estudante, professor, etc;

Modification (modificação): este módulo é tanto uma extensão do módulo de tempo quanto do módulo de ação jurídica. O módulo de tempo é estendido com vários intervalos e momentos descrevendo a eficácia e a vigência dos documentos jurídicos;

Rules (regras): este módulo define um papel central à argumentação e descreve o vocabulário para as regras da LKIF.

Abordagens metodológicas – Na construção da ontologia LKIF, seguiram-se duas abordagens. A primeira é uma abordagem clássica *top-down*, em que uma ontologia superior fornece a estrutura inicial. A ontologia superior é LRI-Core, mas também tem-se investigado o tipo de apoio à modelagem que poderia ser fornecido por outros tipos de ontologias superior. A segunda abordagem pode ser caracterizada como “de meio”, onde grupos de conceitos interdependentes são identificados. Conceitos muito específicos podem ser representados através de scripts ou frames, visando à referencia a **em que o conceito consiste** ou **do que é dependente**, ao invés da definição de *o que ele é*. Ao final, somando em todos os módulos, a

ontologia totalizou 205 classes e 122 propriedades, a qual está completamente disponível em <http://www.estrellaproject.org/lkif-core>.

Características favoráveis ao presente trabalho: trata-se de um grande projeto, envolvendo muitas pessoas, visando a obter uma ontologia muito bem detalhada. O estudo parte de conceitos abstratos, filosóficos e epistemológicos, passando pelo conhecimento da operacionalização das leis, para chegar a uma ontologia que serve de núcleo (daí o termo *core* no nome) para qualquer outra ontologia de aplicação legalista.

Características desfavoráveis ao presente trabalho: por se tratar de uma ontologia de alto nível, não pode ser diretamente utilizada no presente trabalho, sendo que, a sua utilização, como ponto de partida para a ontologia de aplicação aqui almejada, pode ser custosa, por exigir o conhecimento de toda a LKIF antes de estender para o domínio em questão.

2.2.10 Câmara Júnior

O trabalho de Júnior (2007) objetiva propor um modelo de indexação automática de acórdãos, baseado em PLN e no tesouro do CJF, avaliando o seu desempenho, quanto aos índices de precisão e revocação de pesquisas textuais, em relação aos índices correlatos da indexação manual.

A metodologia parte de um processo de treinamento onde um corpus de 3.340 acórdãos do TJDFR é pré-processado através de técnicas de análise morfológica e sintática com intervenção de usuário especialista, sendo que ao final as unidades léxicas geradas são comparadas com o tesouro do CJF, visando a identificar aquelas que tem relevância no domínio.

A última tarefa é a comparação dos índices gerados manualmente e automaticamente. Além disso, os resultados de pesquisas realizadas nas duas bases de dados, com o mesmo motor de busca permitem avaliar, com suporte nos conceitos de revocação e precisão, a qualidade da indexação.

Características favoráveis ao presente trabalho: uso de PLN visando a obter sentenças candidatas e de tesouro jurídico visando à aferição da relevância das sentenças obtidas;

Características desfavoráveis ao presente trabalho: as técnicas de PLN utilizadas não possibilitam a conceitualização dos índices gerados para os acórdãos.

2.2.11 Zahra

“Pode uma ontologia na área de saúde ser construída de maneira semiautomática a partir de textos em português disponibilizados pelos profissionais ou disponíveis na Web?” Esta é a principal questão que norteia o trabalho de Zahra (2009). Dada a grande quantidade de textos disponíveis, espera-se que seja possível utilizar métodos que permitam construir uma ontologia que trate de uma parte razoável do domínio. Porém, a precisão e consistência não podem ser garantidas com métodos automatizados, sendo necessário sempre o pós-processamento por seres humanos. Assim, o trabalho objetiva o desenvolvimento de uma ferramenta para a construção semiautomática de ontologias de domínio⁴ a partir de textos em português.

A ferramenta proposta foi desenvolvida tendo com base a ferramenta para construção semiautomática de ontologias a partir de textos em inglês Texto-to-Onto; as medidas frequência, tfidf e entropia; e a ferramenta de anotação linguística TreeTagger.

Características favoráveis ao presente trabalho: Uso de vocabulário controlado específico da área em questão, visando a melhorar a relevância dos termos extraídos. **Termos ou sentenças extraídas possibilitam uma conceitualização dos documentos.** A fase de extração independe de interferência humana.

Características desfavoráveis ao presente trabalho: as ontologias resultantes são de domínio. No presente trabalho necessita-se de uma ontologia de aplicação. Segundo Guarino (1998), uma ontologia de aplicação pode ser derivada de uma de domínio, o que frequentemente ocorre em conjunto com a especialização de uma ontologia de tarefas. Assim, a utilização de uma ontologia de domínio resultante da ferramenta descrita em Zahra (2009), dada a provável generalidade dos conceitos que seriam extraídos, provavelmente demandaria da construção de uma ontologia de tarefas e da especialização de ambas em uma nova ontologia de aplicação. Estas prováveis situações, somadas ao fato de que o projeto foi experimentado em um domínio bastante distante da jurisprudência, sugerem que a extensão do trabalho de Zahra (2009) são de risco considerável para os objetivos aqui traçados.

2.2.12 Ramos Júnior e Rover

No Brasil, o uso de ontologias é relativamente recente, há poucos projetos existentes e com pouca aplicação prática em comparação a outros países. Assim, o artigo de Júnior e Rover

⁴**Ontologias de Domínio** descrevem o vocabulário relacionado a domínios genéricos (como medicina ou direito), através da especialização dos termos introduzidos em uma ontologia de alto nível (GUARINO, 1998).

(2009) se propõe a expor a importância do uso de ontologias na sociedade do conhecimento, além de apontar diversos exemplos de sua aplicação na área do direito e do governo eletrônico, tanto em âmbito nacional quanto internacional.

Comenta-se sobre vários projetos de desenvolvimento de ontologias legais sendo desenvolvidas na Europa e nos Estados Unidos da América. Destaca-se alguns projetos sobre ontologias jurídicas e voltadas para o governo eletrônico no estado de Santa Catarina, com base na experiência dos autores como pesquisadores em alguns destes projetos.

Evidencia-se a importância do uso de ontologias no direito e principalmente no âmbito do governo eletrônico porque são mais um instrumento eficiente à disposição do governo que poderá ser utilizado na busca pela eficiência, por facilitar o acesso ao conhecimento jurídico aos cidadãos leigos, permitir a recuperação de informações e documentos na web semântica, promovendo a inter-operabilidade e troca de informações entre os órgãos públicos, auxiliando na organização, recuperação e gestão do conhecimento público.

Características favoráveis ao presente trabalho: trata-se um bom apanhado do estado da arte em projetos de ontologias em âmbitos nacional e internacional.

2.2.13 Conclusão

Os trabalhos estudados introduziram várias ideias, artefatos e abordagens, desempenhando assim um importante papel na criação de um arcabouço de técnicas, métodos ou ferramentas, as quais agora servem de base teórica para o desenvolvimento deste projeto. Contribuindo na constituição deste arcabouço podem-se citar: as aplicações de RI e PLN, visando à extração de conhecimento; o uso de vocabulários controlados e tesouros, visando a um aprendizado a priori; métodos para construção de ontologias de forma manual ou semiautomáticas; ontologias de alto nível. Resumindo, o Quadro 1 apresenta as principais técnicas utilizadas nos trabalhos revisados.

Técnica	Subseção											
	2.2.1	2.2.2	2.2.3	2.2.4	2.2.5	2.2.6	2.2.7	2.2.8	2.2.9	2.2.10	2.2.11	2.2.12
Análise Documentária						X						
Classificação Temática				X	X							
Dicionário Conceitual					X							
Entropia											X	
Ontologia							X	X	X		X	X
PLN							X			X		
Raciocínio Baseado em Casos		X	X									
Redes Bayesianas de Crenças	X											
Tesouro jurídico	X	X	X					X		X		
Texto-to-Onto											X	
TFIDF											X	
TreeTagger											X	
Vizinho-mais-próximo		X	X									
Vocabulário controlado				X								
WordNet							X					

Quadro 1: Técnicas utilizadas nos trabalhos correlatos

onde:

Técnica lista as técnicas, ferramentas, artefatos ou algoritmos utilizados na solução das problemas abordados nos trabalhos;

Subseção dispõem em colunas a numeração de cada subseção onde se apresentou o resumo do trabalho;

x indica que a técnica citada na respectiva linha foi utilizada no trabalho apresentado na subseção cujo número consta na respectiva coluna.

Entretanto, os objetivos traçados para o presente trabalho são bastante específicos, cujo alcance não seria possível através da extensão de algum dos trabalhos estudados, podendo-se citar como fatores impeditivos de adoção: métodos que não geram classificação ou conceitualização dos documentos, quando muito tratam os documentos de maneira atômica, visando à busca de similaridade entre eles; métodos com forte dependência de interferência de usuários; métodos que geram terminologias referentes ao campo do direito, mas que não se aproximam do domínio da jurisprudência ou são insuficientes para uma aplicação de gestão de conhecimentos deste ramo. Assim, no presente trabalho pretende-se partir do citado arcabouço para propor métodos para alcançar os objetivos traçados.

2.3 MINERAÇÃO DE TEXTOS

A Mineração de Textos se baseia em técnicas de Inteligência Artificial, sendo as suas tarefas classificadas como métodos descritivos que, segundo Fayyad (1996), buscam padrões humanamente compreensíveis para descrever os dados em tarefas como Descoberta de Regras de Associação, Agrupamento e Descoberta de Padrões Sequenciais.

Os Métodos Descritivos se baseiam em aprendizagem não-supervisionada, onde, segundo Russel e Norvig (2009), não há uma base inicial de exemplos rotulados e busca-se, capturar uma organização inerente dos dados.

Berry e Castellanos (2008) apresentam a Mineração de Textos como um conjunto de abordagens para organizar, classificar, rotular, descobrir tendências e extrair relevantes informações de textos, organizadas em três áreas:

Agrupamento e Classificação: compreende técnicas que visam ao agrupamento de documentos semelhantes e a atribuição de classes que os identifiquem através de, por exemplo: redução de textos; descoberta de similaridade entre palavras; ponderação de relevância de termos ou sentenças; seleção de características baseada em pontuação de relevância de sentenças, etc.;

Extração e Recuperação de Informações: compreende técnicas para se extrair sentenças e informações relevantes de textos visando, por exemplo: à rotulação automática de grupos baseada no conteúdo dos documentos; à descoberta de tópicos importantes em documentos baseado em métodos de categorização de conteúdo; à extração de termos relevantes de corpus visando a gerar dados de treinamento para alimentar algoritmos de aprendizagem de máquina ou bases de conhecimento;

Detecção de Tendências: compreende técnicas para inferência de tendências a partir da mineração em logs de consultas ou textos visando a obter informações detalhadas sobre características linguísticas ou estatísticas, geração de dados de testes ou treinamento, visualização e avaliação. Tais informações podem ser utilizadas para o aprimoramento de sistemas de busca.

2.3.1 Recuperação de Informações

O principal alvo de Recuperação de Informações, RI, hoje, são os motores de busca, baseado em técnicas e ferramentas tais como: consultas baseadas em índices; pré-processamento

e indexação de textos; pesquisa em dicionários; processamento de consultas com erros ortográficos; compressão de dicionários e índices.

As buscas tradicionais se baseiam na Recuperação Booleana para a qual um documento só dispõe de duas possibilidades: ou o documento satisfaz totalmente a consulta ou não. Entretanto a RI possibilita estender o resultado das pesquisas para se basear em quanto o documento se aproxima da consulta, através de métodos para ponderação de relevância de termos e pontuação de documentos, visando a obter resultados ordenados por pontuação.

Além da busca ponderada por pontuação, mais uma forma de se disponibilizar avanços nos motores de busca se dá através da expansão de consulta. Visando a um aumento na probabilidade de se recuperar documentos relevantes, os métodos de RI pretendem expandir ou reformular os termos contidos na consulta fornecida pelo usuário, através de pesquisa de similaridade em vocabulários controlados (pré-existentes ou gerados automaticamente), de técnicas de correção ortográfica ou de retorno dos próprios usuários a respeito da relevância dos documentos retornados.

A RI pode ser associada ao Aprendizado de Máquina visando ao agrupamento ou à classificação de documentos através de métodos mais adequados e específicos para textos.

Na base de todas as aplicações de RI encontram-se técnicas de pré-processamento de termos, extração de sentenças, processamento de vocabulários controlados, cálculo de relevância, processamento linguístico e etc. Nas subseções a seguir serão revisadas algumas técnicas que visam a abordar estes desafios que estão no núcleo da RI, mais notadamente aquelas que serão utilizadas nos métodos propostos para o presente trabalho, conforme o capítulo 3, página 76.

2.3.2 Extração de Sentenças

Na extração de sentenças, segundo Berry e Castellanos (2008), técnicas clássicas, que consideram *ocorrência e localização* de palavras bem como termos de *sinalização*, podem ser utilizadas para gerar um glossário de palavras-chave a ser utilizado na geração de *score* para o cálculo da relevância da sentença.

A *Ocorrência* baseia-se na intuição de que termos que aparecem frequentemente em um texto específico, mas não são comuns no conjunto de textos correlatos, podem ser de grande relevância para o primeiro. A relevância pode ser aferida através de cálculos matemáticos ou de busca em tabelas ou vocabulários controlados com termos relevantes para o domínio. A característica de *Ocorrência* pode estar vinculada à característica de *Localização*, o que significa que o processamento da ocorrência se dará exclusivamente com as sentenças contidas em um

determinado local do documento.

A *Localização* de termos pode denotar a importância de uma sentença, além de indicar o método de localizar um conceito em um texto. Segundo Edmundson (1968) o título e o cabeçalho são locais importantes de um documento, contendo, assim, termos relevantes. Mas Berry e Castellanos (2008) acrescentam que também seções e seus respectivos títulos podem conter termos relevantes, merecendo então a abrangência em uma captura de termos por localização.

Edmundson (1968) propôs a hipótese de que a presença de palavras ou frases pragmáticas podem afetar a relevância de uma sentença. Como exposto por Berry e Castellanos (2008), os termos podem ser classificados como *bonus* (que são positivamente relevantes) ou *stigma* (que são negativamente relevantes), são dependentes do domínio em questão e devem ser levantados manualmente passando a constar de um glossário de termos-chave a ser utilizado na extração das sentenças. Esta é a definição da *Sinalização*, a qual consiste no conjunto de termos, algarismos, letras ou caracteres que podem sinalizar a existência (*bonus*) de um conceito composto pelos próprios sinais ou por outros que estejam no seu entorno. A característica de sinalização pode ocorrer em conjunto com a característica de localização, o que significa que um símbolo pode ser relevante apenas em um local específico do documento.

Quando o sinal consiste em um termo, a sua manipulação pode depender de pré-processamento linguístico (remoção de acentuação, redução em gênero, número ou tempo verbal e etc.). O termo de sinalização pode ser também negativo, ou seja, a sua ausência (*stigma*) pode indicar ou limitar a ocorrência de um conceito. Por exemplo, na busca por indivíduos para o conceito de Comarca a sentença “COMARCA DE ALMIRANTE TAMANDARÉ VARA CRIMINAL” pode ser definida por todos os termos subsequentes ao *bonus* COMARCA porém limitados ao surgimento do *stigma* VARA.

Já quando o sinal consiste em algarismos, letras ou sinais a sua busca pode se dar através de uma expressão regular que represente a sua sintaxe. Como exemplo, no anexo A, há um acórdão, conforme aqueles que estão no foco deste trabalho, onde podem-se identificar os quesitos enumerados por Berry e Castellanos (2008), facilitando assim o seu entendimento, a saber:

Localização: o cabeçalho do documento contém várias sentenças importantes sobre o recurso interposto, bem como as suas partes; o local da seção Ementa contém possíveis informações de Fato, Assunto e Voto; o local da seção Decisão contém informações de Voto;

Sinalização: ao longo do documento encontram-se vários termos *bonus* que sinalizam a possível presença de conceitos esperados, tais como Legislação, Voto, Recurso e Papéis;

Ocorrência: a seção Ementa contém sentenças pressupostamente relevantes para resumir o conteúdo do documento, as quais trazem implícitos os conceitos de Fato e Assunto. Assim, os termos que são relevantes para o documento podem consistir em fatos, enquanto que aqueles que não são relevantes para o documento, mas o são para o domínio da jurisprudência, podem consistir em assuntos.

Este exemplo demonstra que a adoção de técnicas de Extração de Sentenças é fundamental na indexação automática de textos, inclusive no caso de acórdãos. A seguir são abordadas técnicas que auxiliam as tarefas de extração em textos.

2.3.3 Análise de Expressões Regulares

A Análise de Expressões Regulares é uma técnica utilizada para o reconhecimento de padrões pré-definidos em cadeias de caracteres. As expressões regulares consistem em uma maneira de descrever um conjunto de strings com base em características comuns. Elas podem ser usadas para pesquisar, editar ou manipular texto e dados. Existe uma sintaxe específica para criar expressões regulares, que vai além da sintaxe normal das linguagens de programação (NETWORK, 2010).

Cada expressão é, segundo Jargas (2009), uma composição de símbolos, caracteres com funções especiais, chamados meta-caracteres que, agrupados entre si e com *strings* literais, formam uma sequência, uma expressão. Cada meta-caractere desempenha diferentes papéis nas expressões, tais como, classe de caracteres, quantificador, grupo de captura e outros. Desta forma, além de padrões de termos ou sentenças, as expressões regulares tornam-se muito úteis para identificação de informações formatadas, tais como data, horário, endereço de e-mail, dados entre *tags* HTML, números de documentos, leis e etc. Cada expressão pode ser testada em textos, retornando sucesso caso estes obedeçam exatamente a todas as suas condições.

Vários sistemas oferecem suporte a expressões regulares, tais como editores de texto (ex. Notepad++), linguagens de programação (ex. Java, Pearl), sistemas gerenciadores de bases de dados (ex. PostgreSQL) e comandos de sistemas de arquivos (ex. Grep).

Segundo Network (2010) há várias versões de expressões regulares, presentes, por exemplo, no Grep, Perl, Tcl, Python, PHP, e AWK, cada uma com peculiaridades da implementação. No Java2 está disponível a biblioteca **RegEx**, a qual implementa uma sintaxe de expressões regulares similar à do Perl, visando à análise em sistemas desenvolvidos através daquela linguagem.

A forma mais básica de expressão são as **Strings Literais**, que visam à busca de um con-

junto de caracteres na sua forma literal. Por exemplo, a expressão “recurso” retornaria sucesso em uma string ou texto onde fosse encontrada a palavra *recurso*. Demonstrando este exemplo em Java, tem-se:

```
if (Pattern.matches("recurso", token[i]))
```

onde:

Pattern é uma classe da biblioteca RegEx que visa a compilar e manipular uma expressão regular (NETWORK, 2011);

matches é um método que compila a expressão, inserida no primeiro parâmetro, e a compara com a string, inserida no segundo parâmetro, retornando um resultado *booleano* que indica se na string foi encontrado o padrão da expressão compilada;

Uma expressão pode incluir um conjunto de caracteres especiais que afetam o padrão procurado no texto, os **Meta-caracteres**, os quais, na RegEx, segundo Network (2010), consistem nos seguintes: ([{ \ ^ - \$ |] }) & ? * + . .

Se no exemplo acima fosse desejável garantir que o padrão incluísse ocorrências onde a palavra está no plural, poder-se ia adicionar o meta-caractere ‘.’ ao final da expressão, o qual consiste em um curinga para qualquer caractere, desta forma o retorno seria verdadeiro também para o termo *recursos*. Se um meta-caractere for desejado em uma expressão no seu sentido literal, basta prefixá-lo com uma barra invertida (\).

Alguns dos meta-caracteres são utilizados sempre entre os símbolos de colchetes e tem a finalidade de especificar quais caracteres são procurados em um determinado padrão, sendo por isso chamados de **Classes de Caracteres**, os quais, na RegEx, consistem em: [^ - &] . Segundo Network (2010), as Classes de Caracteres podem ser sumarizadas conforme o Quadro 2.

Fonte: (NETWORK, 2010) – tradução livre do autor

Estrutura	Padrão especificado	Operação	Equivalência
$[abc]$	$a, b, \text{ ou } c$	classe simples	
$[\hat{abc}]$	qualquer caractere exceto $a, b, \text{ e } c$	negação	
$[a-zA-Z]$	caracteres de a a z ou de A a Z , inclusive	faixa	
$[a-d[m-p]]$	caracteres de a a d ou de m a p	união	$[a-dm-p]$
$[a-z\&\&[def]]$	$d, e, \text{ ou } f$	intersecção	$[def]$
$[a-z\&\&[\hat{bc}]]$	caracteres de a a z , exceto b e c	subtração	$[a[d-z]]$
$[a-z\&\&[\hat{m-p}]]$	caracteres de a a z , exceto aqueles entre m e p , inclusive	subtração	$[a-lq-z]$

Quadro 2: Estrutura de Classes de Caracteres

Assim, exemplificando as estruturas apresentadas no quadro acima tem-se as seguintes expressões regulares, com os seus resultados:

“**[gp]ato**”: o retorno seria verdadeiro ao se encontrar palavras como *gato* ou *pato*. Entretanto, seria falso para a palavra *mato*;

“**[\^gp]ato**”: o retorno seria **falso** ao se encontrar as palavras *gato* e *pato*. Entretanto, seria verdadeiro ao encontrar palavras como *mato* ou *jato*;

“**[g-j]ato**”: o retorno seria verdadeiro ao se encontrar palavras como *gato* ou *jato* e falso para palavras como *mato* ou *pato*;

“**[g-j[n-p]]ato**”: o retorno seria verdadeiro ao se encontrar palavras como *gato* ou *pato* e falso para *mato* ou *rato*;

“**[g-p&&[j-n]]ato**”: o retorno seria verdadeiro ao se encontrar palavras como *jato* ou *mato* e falso para *gato* ou *pato*;

“**[g-p&&[\^j-n]]ato**”: o retorno seria verdadeiro ao se encontrar palavras como *gato* ou *pato* e falso para *jato* ou *mato*;

Para algumas Classes de Caracteres de uso muito frequente existem atalhos, que visam a identificar classes através de simples *scape sequences*, isto é, letras antecedidas de barra invertida. São as **Classes de Caracteres Predefinidas**, as quais podem ser resumidas através do Quadro 3.

Fonte: (NETWORK, 2010) – tradução livre do autor

<i>Scape</i>	Padrão especificado	Equivalência
<code>\d</code>	um algarismo de 0 a 9	$[0 - 9]$
<code>\D</code>	um caractere diferente de algarismo	$[\^0 - 9]$
<code>\s</code>	um caractere de espaço (espaço em branco, tabulação, quebra de linha, etc.)	$[\ \t\n\ \x0B\f\r]$
<code>\S</code>	um caractere que não produz espaço	$[\^\s]$
<code>\w</code>	um caractere utilizado para compor palavras (letras, algarismos e ‘_’)	$[a - zA - Z0 - 9_]$
<code>\W</code>	um caractere não utilizado na composição de palavras	$[\^\w]$

Quadro 3: Classes de Caracteres Predefinidas

Assim, exemplificando as *scape sequences* apresentadas no quadro acima tem-se as seguintes expressões regulares, com os seus resultados:

“(mesa)\d” : retornaria verdadeiro para **mesa2** e falso para **mesas**;

“(mesa)\D” : retornaria verdadeiro para **mesas** e falso para **mesa2**;

“(mesa)\s\d” : retornaria verdadeiro para **mesa 2** e falso para **mesa-2**;

“(mesa)\S\d” : retornaria verdadeiro para **mesa-2** e falso para **mesa 2**;

“(mesa)\w” : retornaria verdadeiro para **mesa2** e para **mesas** e falso para **mesa?**;

“(mesa)\W” : retornaria verdadeiro para **mesa?** e falso para **mesa2** e para **mesas**;

Alguns meta-caracteres são utilizados para especificar a quantidade de ocorrências de um padrão em um texto. São os **Quantificadores**, que consistem nos caracteres { ? * + }, podendo ser resumidos através do Quadro 4.

Fonte: (NETWORK, 2010) – tradução livre do autor

Estrutura	Padrão especificado
$X?$	X , uma vez ou nenhuma
$X*$	X , nenhuma ou mais vezes
$X+$	X , uma ou mais vezes
$X\{n\}$	X , exatamente n vezes
$X\{n,\}$	X , pelo menos n vezes
$X\{n,m\}$	X , no mínimo n e no máximo m vezes

Quadro 4: Meta-caracteres Quantificadores

Assim, exemplificando as estruturas apresentadas no quadro acima tem-se as seguintes expressões regulares, com os seus resultados:

“(as)?” : submetido ao termo **asas**, retornaria verdadeiro três vezes – uma para cada sílaba (ambas “as”) e uma terceira para a posição vazia ao final da string, visto que o meta-caractere ‘?’ denota a ocorrência do padrão uma vez ou nenhuma;

“(as)*” : submetido ao termo **asas**, retornaria verdadeiro duas vezes – uma para as duas sílabas idênticas “as” e uma segunda para a posição vazia ao final da string, visto que o meta-caractere ‘*’ denota a ocorrência do padrão várias vezes ou nenhuma;

“(as)+” : submetido ao termo **asas**, retornaria verdadeiro uma vez, exclusivamente para as duas sílabas idênticas “as”, visto que o meta-caractere ‘+’ denota a ocorrência do padrão uma ou várias vezes;

“(as){1}” : retornaria verdadeiro para o termo **as** e falso para **asas**;

“(as){1,}” : retornaria verdadeiro para os termos **as** e **asas**;

“(as){2,3}” : retornaria verdadeiro para o termo **asas** e falso para **as**;

Notar que nos exemplos acima os literais “mesa” e “as” apareceram sempre encerrados entre parêntesis. Isto porque, sem os parêntesis, os quantificadores seriam aplicados somente à última letra do literal. Para que os quantificadores fossem aplicados ao literal todo, estes foram encerrados entre parêntesis, os quais, assim, desempenham o papel de **Grupo de Captura** que é, segundo Network (2010), uma forma de tratar múltiplos caracteres de maneira singular. Da mesma maneira, é possível combinar Caracteres de Classe com Quantificadores, por exemplo, a expressão “[as]{2}” retornaria verdadeiro para **as**, **sa**, **aa**, e **ss**.

Ainda em termos de combinação, é possível também a associação de Quantificadores a Classes de Caracteres Predefinidas. Por exemplo, a expressão “\d{1,2}/\d{1,2}/\d{2,4}” retornaria verdadeiro para datas como **01/01/2011** ou **1/1/11**.

As estruturas apresentadas acima formam apenas a base da manipulação de expressões regulares, sendo que este assunto ainda possibilita um avanço muito mais profundo permitindo, segundo Friedl (2006), códigos de processamento de textos complexos (mas sutis) os quais podem poupar tempo e trabalho, possibilitando a construção de soluções eficientes para uma ampla gama de problemas.

2.3.4 *Stemming*

Quando os padrões mapeados se baseiam não apenas em sinais mas também em termos linguísticos, é necessário o uso concomitante de ferramentas de padronização dos termos. Uma

técnica muito utilizada é o *Stemming* (Lematização), a qual visa a reduzir os termos à sua raiz.

O *Stemming*, segundo Alvares (2005), pode ser definido como o processo de transformar formas variantes das palavras para uma representação concisa e precisa, denominada *stem* (tronco), o qual consiste em uma sub-cadeia de uma palavra que representa de forma única e não ambígua a própria palavra e as suas variações, possibilitando assim pesquisas mais abrangentes.

Um documento jurídico, por exemplo, pode conter as palavras *processou*, *processos* e *processual* sendo todas derivadas da palavra *processo*, diferenciadas sintaticamente apenas por sufixos. Entretanto, segundo Alvares (2005), há palavras com semelhança gráfica mas significados diversos, como *eminente* e *iminente*, onde a redução do prefixo não deve resultar em um tronco comum.

Como cada linguagem é constituída, no geral, de raízes e inflexões próprias, não é possível a construção de um algoritmo único para todas as línguas, sendo, assim, necessária a criação de pelo menos um algoritmo de *stemming* para cada uma, visando a mapear as características e particularidades de cada linguagem.

Para a língua portuguesa pode-se citar a biblioteca **PortugueseStemmer**, implementação em Java de um algoritmo⁵ que visa ao processamento de termos da língua, desenvolvida por Dias (2005). O uso do algoritmo desta biblioteca se dá através da simples invocação do método **stem**, passando como parâmetro uma string contendo a palavra que se deseja submeter ao *stemmer*, o qual avalia e, se possível, executa a sua redução nas propriedades de gênero, número, advérbio, aumentativo, diminutivo, substantivo, verbo e acentuação. O resultado do processamento é retornado à chamada do método, consistindo no termo reduzido nas propriedades em que havia possibilidade de redução. Alternativamente pode-se invocar o método **removeAccents** parametrizado com a palavra desejada, visando à simples remoção de acentos, sem redução da palavra.

2.3.5 *Stop-Words*

Stop-Words são palavras comuns ou estruturais – artigos, conjunções, pronomes, preposições, advérbios, verbo “ser”, verbo “estar” e etc. – que não tem qualquer relevância para a maioria dos textos, podendo assim ser eliminadas do texto, sem prejuízo à semântica do documento, facilitando o processamento ao reduzir a dimensão dos dados.

A remoção destas palavras não apresenta muita complexidade, pois consiste simplesmente

⁵O algoritmo desenvolvido para a PortugueseStemmer teve como base o artigo A Stemming Algorithm for the Portuguese Language de Vivianne Orenge e Chris Huyck, in Proceedings of the SPIRE, Novembro de 2001.

em varredura das palavras contidas no texto seguida da regravação somente das palavras não consideradas como stop-words, independentemente da língua do texto. Para auxiliar esta tarefa na construção de sistemas, por exemplo, pode-se citar a classe Java **StopwordsEnglish**, disponível em Frank (2010), a qual carrega um arquivo (que deve conter uma lista com as stop-words definidas) em um vetor em memória principal, disponibilizando um método que verifica se a palavra fornecida consta do vetor. Baseado no retorno de tal método, o sistema pode decidir pela omissão ou não da palavra no arquivo de reescrita do texto.

A inteligência desta técnica, na verdade, reside na construção da lista de palavras a serem eliminadas, variando, naturalmente, de língua a língua. A lista pode variar também de acordo com o domínio envolvido, visto que certas palavras podem ser comuns em alguns domínios, mas em outros não. Tanto assim, que a citada classe não traz consigo arquivo algum de stop-words, facultando ao programador a parametrização do arquivo com a lista que melhor se encaixar à sua língua e ao domínio em questão. Para a língua portuguesa, por exemplo, pode-se utilizar a lista disponível em Savoy (2010) a qual dispõe de 392 stop-words, à qual pode-se acrescentar, quando da aplicação, palavras comuns ao domínio abordado, visando a uma maior redução na dimensão do corpus.

2.3.6 Tesouros

A extração de sentenças baseada na ocorrência remete a técnicas de aferição de relevância de termos em documentos ou corpus, geralmente baseadas em cálculos matemáticos ou consultas a vocabulários controlados, sendo que na primeira forma obtém-se uma relevância genérica, baseada tão somente na frequência dos termos, enquanto que, no segundo caso, pode-se obter uma relevância mais específica, baseado na importância do termo para o domínio em questão.

Considerando que muitos dos termos oriundos do corpus de acórdãos possuem uma semântica jurídica bem específica, buscou-se para este trabalho um vocabulário relativo ao domínio da jurisprudência, visando ao quesito *ocorrência* na extração das sentenças. O Tesouro Jurídico da Justiça Federal (CEJ, 1997) é um ótimo exemplo de mapeamento de terminologia relevante ao domínio. Desenvolvido pelo Centro de Estudos Judiciários (CEJ), órgão do CJF, entre os anos de 1993 e 1997, visa à uniformização da terminologia do Direito nas áreas de competência da Justiça Federal, a ser utilizada em sistemas e bases de dados dos tribunais federais ou de outras instituições que tenham alguma interação com tal competência. A última atualização deste tesouro se deu em 13/02/2007, passando então a contar com 8.527 descritores e 1.089 sinônimos.

Tesouro é definido por Motta (1986) como um sistema de vocabulário baseado em concei-

tos, incluindo termos preferidos (descritores), termos não preferidos (não descritores) e suas inter-relações, que se aplica a um determinado ramo do conhecimento e que se destina a controlar a terminologia utilizada para a indexação/recuperação de documentos. Um tesouro se compõem por duas camadas: **léxico**, constituído dos termos pertinentes ao domínio para o qual o tesouro foi construído, e **estrutura**, que estabelece a relação entre os termos tais como sinonímia, hierarquia, etc.

Na camada léxica os termos são classificados, segundo CEJ (1997), como Descritores, que são os eleitos para representar os conceitos na indexação e recuperação de informações, e Não-Descritores, que são aqueles que, embora representem os mesmos conceitos que os descritores, não são autorizados para uso na indexação e recuperação, evitando-se a proliferação de sinônimos que dificultam a recuperação da informação.

Na camada estrutural de um tesouro costuma-se especificar, segundo Motta (1986), três tipos de relações hierárquicas entre termos: as genéricas (representadas pela sigla TG), as específicas (TE) e as demais (TR). TG indica, segundo CEJ (1997), o conceito mais abrangente, do qual o termo subordinado (TE) é um tipo, enquanto TR indica um conceito relacionado, não por equivalência ou hierarquia, mas, por afinidade, sendo tão comumente associados mentalmente que se deve tornar essa relação explícita no tesouro. Além das relações por hierarquia, CEJ (1997) explicita também as relações de equivalência entre descritores e não-descritores, representadas pelas siglas UP e USE. UP indica o não-descritor que se equivale ao corrente descritor, enquanto que USE indica a qual descritor o corrente não-descritor se refere. Como exemplo, foram selecionados alguns descritores inter-relacionados do tesouro do CJF:

COAÇÃO MORAL

UP VIS COMPULSIVA

TG1 COAÇÃO

TE1 COAÇÃO MORAL IRRESISTÍVEL

TR COAÇÃO FÍSICA

VIS COMPULSIVA

USE COAÇÃO MORAL

COAÇÃO FÍSICA

UP VIS ABSOLUTA

TG1 COAÇÃO

TR COAÇÃO MORAL

COAÇÃO

TE1 COAÇÃO FÍSICA

TE1 COAÇÃO MORAL

TE2 COAÇÃO MORAL IRRESISTÍVEL

TR AÇÃO ANULATÓRIA

COAÇÃO MORAL IRRESISTÍVEL

TG1 COAÇÃO MORAL

TG2 COAÇÃO

TR CULPABILIDADE

Deste tesauro podem-se extrair as seguintes informações:

COAÇÃO MORAL é um descritor que se equivale ao não-descritor **VIS COMPULSIVA**, é especialização de **COAÇÃO** e generalização de **COAÇÃO MORAL IRRESISTÍVEL** e pode ser associado a **COAÇÃO FÍSICA**;

VIS COMPULSIVA é um não-descritor que se equivale ao descritor **COAÇÃO MORAL**;

COAÇÃO FÍSICA é um descritor que se equivale ao não-descritor **VIS ABSOLUTA**, é especialização de **COAÇÃO** e pode ser associado a **COAÇÃO MORAL**;

COAÇÃO é um descritor especializado por **COAÇÃO FÍSICA** e por **COAÇÃO MORAL** (o qual, por sua vez, é especializado por **COAÇÃO MORAL IRRESISTÍVEL**) e pode ser associado a **AÇÃO ANULATÓRIA**;

COAÇÃO MORAL IRRESISTÍVEL é um descritor que especializa **COAÇÃO MORAL** (o qual, por sua vez, especializa **COAÇÃO**) e pode ser associado a **CULPABILIDADE**.

Desta maneira, através de siglas, o tesauro demonstra consistir em um texto estruturado capaz de exprimir uma taxonomia de conceitos que definem um determinado domínio, conforme pode-se visualizar através do diagrama de classes, gerado à partir do trecho do tesauro acima, apresentado na Figura 1. Note que no diagrama os arcos não encontram-se nominados, como seria comum na maioria de artefatos deste tipo. Isto se deve ao fato de o tesauro não incluir uma informação que explicita a semântica das relações entre descritores. Assim optou-se aqui por manter arcos inominados, evitando assim a presunção de uma semântica que pode não coincidir com aquelas pretendidas pelos autores do tesauro.

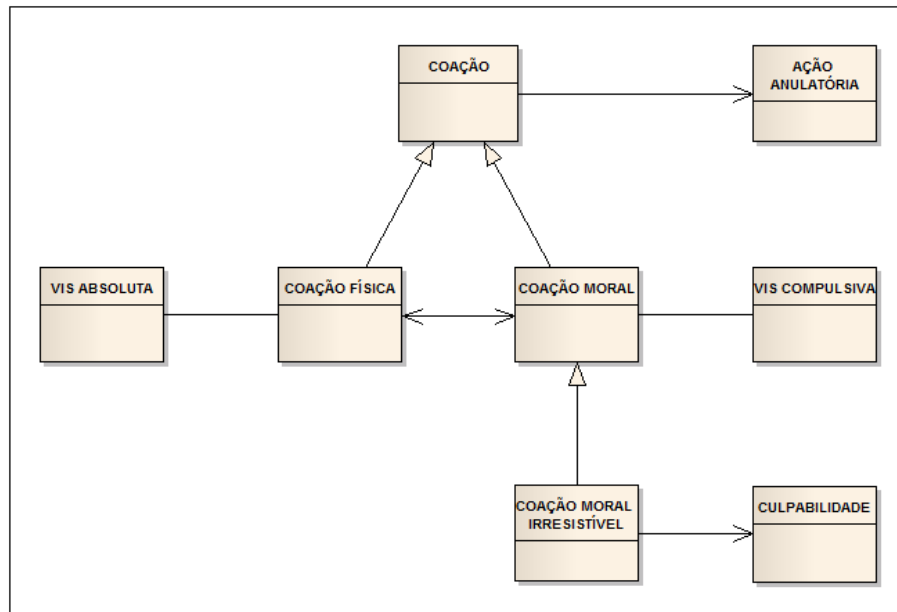


Figura 1: Taxonomia presente no Tesauro do CJF

2.4 GESTÃO DE CONHECIMENTO BASEADO EM ONTOLOGIAS

A Gestão do Conhecimento é atualmente uma ferramenta estratégica para corporações, sejam elas privadas ou públicas. A Tecnologia da Informação desempenha um papel quase que obrigatório neste contexto, suprimindo uma ampla sorte de pesquisas, teorias, abordagens e técnicas que estão originando inúmeros artefatos tecnológicos que visam a solucionar a gestão. O principal desafio está na *Modelagem de Informações*, a qual visa, segundo BAADER et al. (2003), à construção de estruturas de símbolos computacionais para modelagem de alguma parte do mundo real. Tais estruturas de símbolos constituem *bases de informações*, geralmente tratadas em Ciência da Computação através de *bases de dados* ou *bases de conhecimento*, as quais são conferidas em consistência e, geralmente, construídas, consultadas e atualizadas através de linguagens específicas para este propósito.

BAADER et al. (2003) ainda esclarece que a linguagem utilizada na construção do modelo de informações influencia nos tipos de detalhes que são considerados. Por exemplo, modelos de bases de dados relacionais são baseadas em noções convencionais de linguagens de programação tais como *registros* ou resultados baseados em aspectos da implementação. Já os *modelos conceituais* disponibilizam facilidades mais expressivas para modelar aplicações direta e naturalmente e para estruturar bases de informações, visto que suas linguagens provêm termos semânticos para modelagem e aplicação, tais como entidade e relacionamento (ou até atividade, agente e meta), bem como significado para organização de informações. Os modelos conceituais desempenham importante papel em diversas áreas, conforme apresentadas por Mylopoulos

(1998):

- programas de **Inteligência Artificial** requerem a representação de grandes montantes de conhecimento humano como condição para procederem “inteligentemente”, partindo, muitas vezes, de modelos conceituais, construídos através da utilização de linguagens de representação de conhecimento, tais como redes semânticas – grafos especializados e rotulados com identificadores em linguagem natural – e Lógica de Descrição, que visa a formalizar as redes semânticas;
- o desenvolvimento de **sistemas baseados em bases de dados** costumam ter entre suas fases iniciais a construção de **um esquema em nível conceitual** que determina as necessidades de informação dos usuários, podendo ser inclusive convertido em um esquema de implementação física;
- em geral, a construção de qualquer tipo de *software* parte de uma atividade de levantamento de requisitos, que, hoje em dia, pode consistir em um modelo de requisitos que descreve o relacionamento entre o sistema proposto e o ambiente em que se insere. **O ambiente, no caso, é semelhante a um modelo conceitual;**
- a **Análise e Programação Orientada por Objetos** propõe uma visão dos componentes de *software* (classes/objetos) como modelos de entidades do mundo real, sendo assim as linguagens orientadas a objetos, como Java e UML, fortemente baseadas em modelos conceituais.

Assim, uma importante alegação dos benefícios da abstração do mundo real em modelos conceituais, segundo BAADER et al. (2003), é o seu resultado em modelos estruturados de informações, fáceis de construir e manter. Pesquisas no campo da representação e raciocínio de conhecimento geralmente tem enfoque em métodos para fornecer descrições de alto nível do mundo que podem ser efetivamente utilizadas para criar aplicativos inteligentes. Neste contexto, “inteligentes” se refere à capacidade de um sistema para obter consequências implícitas dos conhecimento explicitamente representados. Tais sistemas são, portanto, caracterizados como Sistemas Baseados em Conhecimento.

2.4.1 Representação de Conhecimento

As abordagens para representação de conhecimento desenvolvidas na década de 1970, costumam ser divididas, segundo BAADER et al. (2003), em duas categorias: *formalismo baseado*

na lógica, que parte do princípio que o Cálculo de Predicados⁶ é capaz de representar fatos sobre o mundo; *representação não-lógica*, geralmente baseada em noções mais cognitivas (por exemplo, redes semânticas e regras representativas inspiradas nas formas de representação de memórias e experiências humanas).

Embora as abordagens não-lógicas tenham sido desenvolvidas visando aplicações específicas, elas acabaram evoluindo para propósitos genéricos. Por outro lado, uma vez que a Lógica de Primeira Ordem⁷ (FOL) provê mecanismos genéricos independentes do domínio onde é utilizada, as abordagens baseadas na lógica foram voltadas a propósitos gerais desde o início.

Em uma abordagem lógica a linguagem de representação sempre consiste em uma variação do cálculo de predicados e o raciocínio implica na verificação de consequência lógica. Já as abordagens não-lógicas podem basear-se no uso de linguagens visuais, sendo o conhecimento representado pelo significado através de estruturas de dados *ad hoc* e o raciocínio realizado por métodos correlatos de manipulação das estruturas. Entre estas abordagens, destacam-se as baseadas em redes: Redes Semânticas e *Frames*.

Devido ao seu enfoque no ser humano, os sistemas baseados em redes foram considerados mais atrativos e eficazes que os sistemas lógicos. Entretanto eles não foram satisfatórios devido à sua falta de precisão semântica. Com isso, surgiu a seguinte questão: *como prover semântica às estruturas de representação, em particular redes semânticas e frames?* Um importante passo nesta direção foi o reconhecimento que os *frames* poderiam exprimir semântica baseado na lógica de primeira ordem através de elementos básicos de representação como predicados unários, denotando conjuntos de indivíduos, e predicados binários, denotando relacionamento entre indivíduos. Entretanto isso não vai de encontro aos requisitos de redes semânticas e *frames* em termos de lógica. Apesar de a lógica consistir em uma base natural para especificação do significado de estruturas, as *frames* e redes semânticas não requerem todos os mecanismos da lógica de primeira ordem, mas de apenas parte deles. Por fim, diferentes características da linguagem de representação poderia levar a diferentes fragmentos da lógica de primeira ordem.

A partir deste contexto se iniciou a pesquisa na área de Lógica de Descrição (DL), sob o rótulo de *sistemas terminológicos* com ênfase na linguagem de representação visando a estabe-

⁶**Cálculo de Predicados** é um sistema de lógica que inclui funções de domínios, tais como números para valores booleanos, sendo suficiente para a maioria das aplicações de lógica matemática, tais como a formalização de aritmética e álgebra. Similarmente, a maioria das aplicações de lógica computacional utiliza ou o cálculo de predicados ou um sistema de lógica formulado a partir do cálculo de predicados. Uma utilização muito importante desta área da matemática é a formalização de semânticas de linguagens de programação, bem como a especificação e verificação de programas (BEN-ARI, 2001).

⁷O Cálculo de Predicados pode, segundo Ben-Ari (2001), também ser chamado por **Lógica de Primeira Ordem**.

lecer a terminologia⁸ básica dos domínios modelados. Posteriormente a ênfase foi dada a conjuntos de construtores para formação de conceitos, dando origem, assim, ao nome **linguagem para definição de conceitos**. Recentemente, depois que a atenção foi deslocada da linguagem para as propriedades dos sistemas lógicos subjacentes (por exemplo, completude e consistência dos métodos de inferência), o termo **Lógica de Descrição** tornou-se popular (BAADER et al., 2003).

A seguir serão abordados os elementos básicos que compõem a representação do conhecimento através de estruturas de rede, servindo de base para introduzir os elementos da DL, bem como do raciocínio.

2.4.1.1 Estruturas de Representação de Conhecimento Baseado em Rede e em DL

Os elementos de uma rede, segundo BAADER et al. (2003), são **nós** – utilizados para representar conceitos (conjuntos ou classes de objetos individuais) – e **arcos** (*links*) – utilizados para demonstrar o relacionamento entre os nós. Conceitos podem ter simples propriedades (atributos) ligados ao nó correspondente. Relacionamentos mais complexos podem ser representados por nós, mas de maneira distinta da representação de conceitos. Como exemplo, tome-se a rede da Figura 2 a qual representa um conhecimento a respeito de pessoas, pais, filhos e etc.

Fonte: (BAADER et al., 2003) – tradução livre do autor

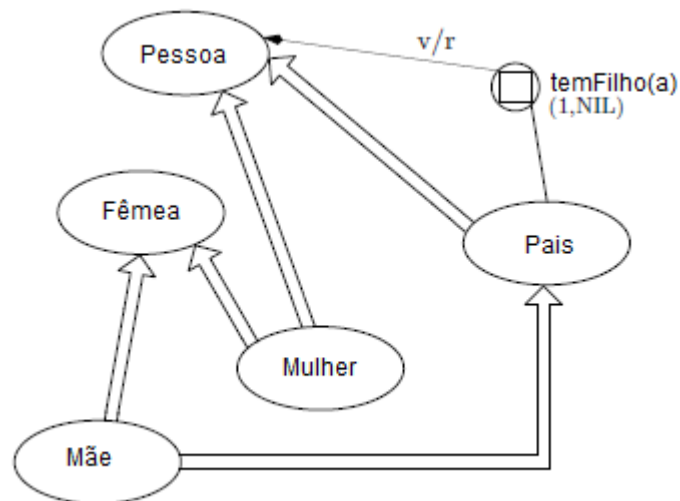


Figura 2: Um exemplo de rede

A estrutura (ou terminologia) da figura pretende representar generalidades e especificidades que há entre os conceitos ali representados. Por exemplo, entre *Mãe* e *Pais* há um arco que

⁸**Terminologia** aqui denota uma estrutura (hierárquica) construída para a representação de um domínio (BAADER et al., 2003).

indica que “mães são pais” – este tipo de relacionamento costuma ser chamado de *IS-A* (É-UM), o qual define a hierarquia que há entre conceitos, provendo assim a herança de propriedades de conceitos genéricos para conceitos específicos.

A DL é capaz de representar outros tipos de relacionamentos entre conceitos, além do *IS-A*. Por exemplo, na figura, o conceito *Pais* tem uma propriedade do tipo “papal”, expressa por um arco que vai do conceito até o nó *temFilho(a)*. O papel tem “restrição de valor” (indicada por v/r), a qual limita o tipo de objeto que pode preencher o papel (no caso um indivíduo de *Pessoa*) e cardinalidade (no caso (1,NIL), onde NIL significa infinito). Assim, a representação do conceito *Pais*, poderia ser lido como “Pais são pessoas que tem, no mínimo, um filho(a), e todos os seus filhos(as) são pessoas”.

Relacionamentos deste tipo podem ser herdados pelos sub-conceitos, por exemplo: o conceito *Mãe*, que é uma fêmea de pais, é descendente dos conceitos *Fêmea* e *Pais*, herdando de *Pais* a ligação com *Pessoa*, através do papel *temFilho(a)*.

Na rede podem haver relacionamentos implícitos. Por exemplo, na Figura 2, depreende-se que se *Mulher* é uma *Pessoa* e uma *Fêmea*, então *Mãe* é uma *Mulher*. Este último relacionamento não está explícito, mas pode ser derivado do fato de *Mãe*, além de ser uma *Fêmea* estar relacionado indiretamente, através do relacionamento com *Pais*, ao conceito *Pessoa*.

Baseado nestas ideias, muitos sistemas foram construídos, necessitando de uma caracterização do significado das estruturas usadas nas representações, bem como do conjunto de inferências que podem ser extraídas dessas estruturas. Assim, uma precisa caracterização do significado de uma rede pode se dar através da definição de uma linguagem para os elementos da estrutura (sintaxe) e da interpretação das *strings* da linguagem (semântica).

A sintaxe DL parte de dois elementos básicos: *Conceitos Atômicos* – designados por símbolos de predicados unários – e *Papéis Atômicos* – designados por símbolos de predicados binários, sendo estes utilizados para expressar relacionamento entre conceitos. Assim os termos são construídos a partir dos símbolos básicos, gerando vários tipos de construtores, por exemplo: *Intersecção de Conceitos* é denotado por $C \sqcap D$ sendo utilizado para restringir o conjunto de indivíduos em questão àqueles que pertencem tanto a *C* quanto a *D*. Nota-se que em DL as expressões de conceitos são livres de variáveis – em lógica de primeira ordem a mesma expressão seria $C(x) \wedge D(x)$, onde x consiste em todos os indivíduos no domínio da interpretação e $C(x)$ é verdadeiro para os indivíduos que pertencem ao conceito *C*. Outra diferenciação que pode ser citada, segundo Brachman e Levesque (2004), é a possibilidade de na DL se ter conceitos não-atômicos, podendo assim possuir relacionamento semântico com outros conceitos.

Uma das principais características da DL, segundo BAADER et al. (2003), residem em construções que definem conceitos por meio de restrições. Um exemplo seria definir um conceito por meio de uma restrição composta pelo quantificador universal e um papel R qualificado pelo conceito C . Assim, a sentença $\forall R.C$ define indivíduos que se relacionam apenas com indivíduos do conceito C por meio de R (ou que não participam da relação definida por R).

Em termos de semântica, conceitos são interpretações de conjuntos teóricos: um *conceito* consiste em um conjunto de indivíduos e *papel* é um conjunto de pares de indivíduos, sendo que o domínio destes conjuntos pode tanto ser definido arbitrariamente, quanto ser considerado infinito. Um domínio infinito está vinculado ao *Pressuposto de Mundo Aberto*, o qual impossibilita que verdades sejam inferidas a partir da inexistência de indivíduos explícitos. Esta característica distingue a DL das linguagens de modelagem de base de dados, onde a ausência de tuplas provoca a presunção de verdades.

Conceitos atômicos são entendidos como subconjuntos de interpretações de domínios, enquanto que a semântica das outras construções é especificada pela definição dos indivíduos envolvidos. Assim, $C \sqcap D$ é o conjunto de indivíduos resultantes da intersecção dos conjuntos de indivíduos C e D , enquanto que $\forall R.C$ é o conjunto de indivíduos que se relacionam com os indivíduos pertencentes ao conjunto C , através do relacionamento R .

Por exemplo, considerando como atômicos os conceitos *Fêmea*, *Pessoa* e *Mulher* e os papéis *temFilho(a)* e *temParenteFemea*, através do uso de operadores de **intersecção**, **união** e **complemento** de conceitos (também conhecidos por, respectivamente, *conjunção de conceito*, *disjunção de conceito* e *negação de conceito*), podem-se descrever conceitos como “pessoas que não são fêmeas” e “indivíduos que são fêmeas ou machos”, através das seguintes expressões:

$$\text{Pessoa} \sqcap \neg \text{Femea} \quad \text{e} \quad \text{Femea} \sqcup \text{Macho}$$

Os papéis podem sofrer restrições através de *quantificadores existenciais* (\exists) e *universais* (\forall), e também, de valores, que possibilitam, por exemplo, a descrição de conceitos como “indivíduos que tem pelo menos um filho(a) fêmea” ou “todos os indivíduos cujos filhos(as) são fêmeas”, através das seguintes expressões, respectivamente:

$$\exists \text{temFilho_a.Femea} \quad \text{e} \quad \forall \text{temFilho_a.Femea}$$

Quantificadores existenciais e restrições de valores visam a caracterizar o relacionamento entre conceitos. Por exemplo, o papel que liga *Pais* a *Pessoa* na Figura 2, página 50, poderia ser

descrito pela seguinte expressão:

$$\exists \text{temFilho_a.Pessoa} \sqcap \forall \text{temFilho_a.Pessoa}$$

Assim, esta expressão caracteriza o conceito de *Pais* como o conjunto de indivíduos onde **pelo menos** um pertence ao conceito Pessoa, mas, ao mesmo tempo, todo filho(a) **deve** ser uma pessoa. Isso impede, por exemplo, que animais, ou coisas inanimadas, tenham filho(a).

Outro importante tipo de restrição de papel, segundo BAADER et al. (2003), é a *restrição de número*, a qual limita a cardinalidade dos conjuntos de indivíduos que preenchem os papéis. Por exemplo, a expressão:

$$(\geq 3 \text{ temFilho_a}) \sqcap (\leq 2 \text{ temParenteFemea})$$

representa o conceito “indivíduos que tem, pelo menos, 3 filhos(as) e, no máximo, 2 parentes fêmeas”.

Além de construções para formação de expressões de conceitos, a DL provê construtores para papéis que podem criar hierarquia entre papéis. A *intersecção* é um exemplo disso, segundo BAADER et al. (2003). Intuitivamente $\text{temFilho_a} \sqcap \text{temParenteFemea}$ denota um papel *temFilha*. Assim, a expressão:

$$\text{Mulher} \sqcap \leq 2 (\text{temFilho_a} \sqcap \text{temParenteFemea})$$

descreve o conceito “uma mulher que tem, no máximo, 2 filhas”.

Por fim, dentre os elementos básicos da DL, deve-se citar a base do *raciocínio*. A inferência mais elementar em expressões de conceitos, segundo BAADER et al. (2003), é a *subsunção*, tipicamente designada por $C \sqsubseteq D$, onde se verifica se o conceito denotado por D é mais geral que aquele denotado por C . Por exemplo, se fosse desejável obter a informação se *Mulher* é subsumida por *Mãe* ($\text{Mulher} \sqsubseteq \text{Mae}$), ou seja, se mulher é um subconjunto de mãe, devem-se verificar os relacionamentos existentes na terminologia, o que pode se obter diretamente nas expressões de conceitos. Uma outra inferência típica em expressões de conceitos é a *Satisfabilidade*, a qual confere se um conceito não denota conceitos vazios. Caso um axioma do conceito resultar em um conjunto vazio, então o conceito todo será considerado insatisfável e, portanto, inconsistente. Estes elementos de raciocínio, na verdade são herança da FOL, porém, segundo Brachman e Levesque (2004) a DL tem um diferencial: a existência de procedimentos de inferência decidíveis capazes de testar se um conceito é satisfável ou não.

2.4.1.2 Aplicação de Conhecimento Representado em DL

Aborda-se em seguida o uso da DL na construção de aplicativos baseados em conhecimento. Em uma base de conhecimento podem-se distinguir dois tipos de conhecimento: “*intensional*” e “*extensional*”, sendo que o primeiro corresponde ao conhecimento geral sobre o domínio em questão, enquanto que o segundo refere-se a um problema em particular. Uma base de conhecimento representada através da DL, segundo BAADER et al. (2003), é analogamente composta de dois componentes: uma *TBox* e uma *ABox*. O *TBox* contém o conhecimento “intensional” na forma de uma **terminologia** ou **taxonomia**, sendo composta por declarações gerais de propriedades de conceitos. Já o *ABox* contém o conhecimento “extensional” ou **afirmativo**, relativo aos indivíduos do domínio.

O **TBox** é composto por declarações que definem a terminologia do domínio. Segundo BAADER et al. (2003), a forma mais básica de declaração é a *Definição* de conceito, através da qual se define um novo conceito atômico baseado em outros previamente definidos. Tal declaração se dá através de um *Axioma de Igualdade* que tem a forma:

$$C \equiv D \quad \text{ou} \quad R \equiv S$$

onde C e D são conceitos e R e S são papéis. Assim, uma igualdade onde o membro da esquerda é um conceito atômico consiste em uma *definição*. Definições são utilizadas para se atribuir nomes simbólicos a descrições complexas. Por exemplo, no axioma:

$$\text{Mae} \equiv \text{Mulher} \sqcap \exists \text{ temFilho_a.Pessoa}$$

definiu-se o conceito de *Mãe* como uma *Mulher* que *tem Filho(a) Pessoa*. Nomes simbólicos podem ser utilizados para encurtar outros axiomas. Assim, na definição do conceito *Pais*, ao invés do complexo axioma:

$$\text{Pais} \equiv (\text{Mulher} \sqcap \exists \text{ temFilho_a.Pessoa}) \sqcup (\text{Homem} \sqcap \exists \text{ temFilho_a.Pessoa})$$

dado que *Pai* é um conceito que foi definido analogamente a *Mãe*, poder-se-ia definir o conceito de *Pais* através de um axioma mais reduzido:

$$\text{Pais} \equiv \text{Mae} \sqcup \text{Pai}$$

Assim, uma *definição* é, segundo BAADER et al. (2003), geralmente interpretada como uma equivalência lógica que provê condições tanto suficientes quanto necessárias para classificar um indivíduo como pertencente a um conceito. Quando não é possível definir completamente um conceito, podem-se reunir as condições somente necessárias para a classificação dos indivíduos, através de um *Axioma de Inclusão* que tem a forma:

$$C \sqsubseteq D \quad \text{ou} \quad R \sqsubseteq S$$

Se o membro esquerdo da inclusão for atômico, então o axioma é chamado de *Especialização*. Por exemplo, para a definição do conceito de *Mulher* o Engenheiro de Conhecimento pode simplesmente declarar que *Mulher* é uma *Pessoa*, através do axioma de especialização:

$$\text{Mulher} \sqsubseteq \text{Pessoa}$$

Um conjunto finito de definições ou especializações é chamado de *terminologia* ou *TBox*, desde que nenhum nome simbólico seja definido mais de uma vez, ou seja, para todo conceito atômico há no máximo um axioma que o contém no lado esquerdo. Assim, uma terminologia para o domínio de relações familiares, conforme exemplificado por BAADER et al. (2003) (e traduzido livremente pelo autor), poderia ser:

$$\begin{aligned} \text{Mulher} &\sqsubseteq \text{Pessoa} \\ \text{Homem} &\equiv \text{Pessoa} \sqcap \neg \text{Mulher} \\ \text{Mae} &\equiv \text{Mulher} \sqcap \exists \text{temFilho_a.Pessoa} \\ \text{Pai} &\equiv \text{Homem} \sqcap \exists \text{temFilho_a.Pessoa} \\ \text{Pais} &\equiv \text{Pai} \sqcup \text{Mae} \\ \text{Avo} &\equiv \text{Mae} \sqcap \exists \text{temFilho_a.Pais} \\ \text{MaeComVariosFilhos_as} &\equiv \text{Mae} \sqcap \geq 3 \text{ temFilho_a} \\ \text{MaeSemFilha} &\equiv \text{Mae} \sqcap \forall \text{temFilho_a.} \neg \text{Mulher} \\ \text{Esposa} &\equiv \text{Mulher} \sqcap \exists \text{temMarido.Homem} \end{aligned}$$

O **ABox** é, segundo BAADER et al. (2003), composto de afirmações sobre indivíduos relativos ao domínio em questão, sendo geralmente chamadas de *afirmações de associação*. Como exemplo, a afirmação:

$$\neg \text{Mulher} \sqcap \text{Pessoa} (\text{ELROND})$$

define que o indivíduo ELROND é uma pessoa e não é mulher. Dada a definição de homem, na terminologia acima, pode-se derivar desta afirmação que ELROND é uma instância do conceito *Homem*. Assim, afirmações deste tipo são chamadas de *Afirmações de Conceitos*. Exemplificando ainda, a afirmação:

temFilho_a(*ELROND*,*ARWEN*)

especifica que ELROND tem um filho(a) que é ARWEN, consistindo, este tipo de afirmação, em uma *Afirmação de Papel*.

2.4.2 Ontologias

De um ponto de vista filosófico, uma ontologia é um sistema de categorias, independente de linguagem, que resulta em uma visão específica do mundo. Em Inteligência Artificial, ontologia consiste em um artefato de engenharia, dependente de linguagem, constituído das seguintes partes (GUARINO, 1998):

- vocabulário, utilizado para descrever uma realidade específica;
- afirmações explícitas presumindo o significado pretendido para cada vocábulo, geralmente construídas através da teoria da FOL. Desta forma os vocábulos nomeiam predicados unários e binários, sendo então chamados, respectivamente, de conceito e relação;
- relações de subsunção que descrevem uma hierarquia de conceitos;
- axiomas, informações básicas ou definições de conceitos. Alguns fornecem informações gerais sem definir completamente um conceito (RUSSELL; NORVIG, 2009).

As afirmações e os axiomas podem ser construídos, segundo Brachman e Levesque (2004), através de qualquer linguagem, até mesmo a linguagem pátria, o que, obviamente, não é produtivo, por não favorecer uma descrição formal e acabar por tornar a lógica muito particular ao local. A Lógica de Descrição, consiste em uma forma adequada para descrever afirmações e axiomas de uma ontologia.

Por fim, em se tratando de um artefato descritivo, baseado em padrões de linguagem conceitual, uma ontologia bem definida tem a característica de compartilhamento de conhecimento, possibilitando a sua reutilização, ou extensão, em outros projetos baseados em bases de conhecimentos. Além disso, o uso de uma ontologia pode facilitar a customização do sistema a

particularidades de aplicações do mesmo domínio ou de domínios paralelos. Uma ontologia facilita a inter-operabilidade entre sistemas diversos, porém relativos a um mesmo domínio.

2.4.3 Ontologias e a Web Semântica

Ontologias figuram na Web Semântica, segundo Heflin (2004), como um caminho para a representação da semântica dos documentos possibilitando o uso da semântica por aplicativos web e agentes inteligentes. Ontologias demonstram ser muito úteis para estruturação e definição do significado de metadados coletados e padronizados, possibilitando que os aplicativos atuem “inteligentemente” no sentido de trabalhar com maior acurácia em um nível conceitual humano.

Nesta subseção se aborda o estado da arte em termos de gestão de ontologias na web, desde o seu princípio, posto que o atual estado é resultante de uma sequência de estudos e definições interligados da World Wide Web Consortium (W3C).

2.4.3.1 Representação de Informações na Web

A abordagem da W3C em representação de informações se iniciou na **XML**, eXtensible Markup Language, a partir da estrutura baseada em marcadores (*tags*) do HTML. Documentos XML, segundo Bray et al. (2000), são compostos de unidades de armazenamento chamadas *entidades*, que contêm dados analisados ou não. Dados analisados são compostos por caracteres, alguns dos quais formando *strings*, e outros formando *marcadores*. Por exemplo:

```
<Instituicao>
  <NomeCompleto>
    PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
    E INFORMÁTICA INDUSTRIAL
  </NomeCompleto>
  <NomeCurto>CPGEI</NomeCurto>
</Instituicao>
```

Um marcador codifica uma descrição do leiaute de armazenagem do documento bem como a sua estrutura lógica. XML fornece um mecanismo para impor restrições sobre o esquema de armazenagem e estrutura lógica.

O padrão XML favorece o intercâmbio de dados entre parceiros, uma vez que eles tenham combinado previamente as definições dos dados. Entretanto, a sua falta de semântica impede, segundo Heflin (2004), que as máquinas executem esta tarefa de forma confiável quando do

surgimento de novos vocábulos XML. Além disso, o mesmo termo pode ser usado com diferentes (às vezes sutilmente) significados em contextos diversos, enquanto termos diferentes podem ser usados para itens que têm o mesmo significado. Por isso, os padrões RDF e RDF Schema iniciaram a abordagem deste problema ao permitir que semânticas simples sejam associadas com identificadores.

A Resource Description Framework, (**RDF**), é uma *framework* baseada no XML criada para, segundo Klyne e Carroll (2004), representar informações na web, através de uma sintaxe abstrata, que reflete um modelo simples de dados baseado em grafos, e de uma semântica formal, com uma noção de implicação rigorosamente definida que provê uma base para deduções bem fundamentadas em dados RDF. Assim, este padrão provê uma forma de expressar simples declarações a respeito de recursos, utilizando propriedades nominadas e valores.

Cada expressão RDF tem a estrutura de uma *tripla*, consistindo em um *sujeito*, um *predicado* e um *objeto*, a qual pode ser representada por um grafo onde cada tripla é composta por uma ligação nó-arco-nó, como na Figura 3.

Fonte: (KLYNE; CARROLL, 2004) – tradução livre do autor

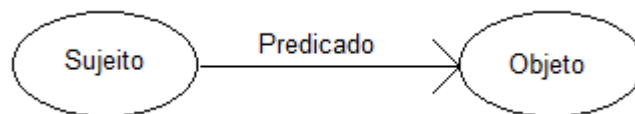


Figura 3: Tripla de afirmação RDF

Cada tripla representa uma afirmação de relação entre coisas denotadas pelos nós. A relação é denotada pelo predicado, sendo o sentido do seu arco sempre do sujeito para o objeto. A identificação de cada nó ou arco pode se dar através de uma URI⁹, um literal ou em branco, sendo que uma URI ou um literal representam aquilo a que se refere o nó, enquanto que um nó em branco possibilita a existência de várias triplas relacionadas a um único nó raiz. A Figura 4, ilustra um grafo RDF composto por nós e arcos identificados por URIs e literais.

⁹Uma Uniform Resource Identifier (URI) fornece um meio simples e extensível para identificar um recurso. A especificação da sintaxe e semântica da URI é derivada de conceitos introduzidos pela iniciativa de informação global da World Wide Web, cuja utilização destes identificadores data de 1990 (BERNERS-LEE; FIELDING; MASINTER, 2005).



Figura 4: Grafo RDF

Fonte: (MANOLA; MILLER, 2004)

Onde:

Eric Miller é um objeto literal correspondente a um indivíduo;

http://www.w3.org/People/EM/contact#me é uma URI que identifica um sujeito;

http://www.w3.org/2000/10/swap/pim/contact#fullName é uma URI que identifica o predicado que tem a função de relacionar um objeto literal (*Eric Miller*, no exemplo) ao sujeito *contact#me*;

Person é um conceito para tipagem representado no grafo pelo objeto URI

<http://www.w3.org/2000/10/swap/pim/contact#Person>;

mailbox é uma propriedade representada pelo predicado URI

<http://www.w3.org/2000/10/swap/pim/contact#mailbox>

mailto:em@w3.org é o valor da propriedade mailbox consistindo no grafo de um objeto literal;

Dr. é um indivíduo representado por um objeto literal que visa a valorar a propriedade *personalTitle* através do predicado URI <http://www.w3.org/2000/10/swap/pim/contact#personalTitle>

Para representar e armazenar um grafo, segundo Manola e Miller (2004), RDF baseia-se em uma sintaxe baseada em XML – chamada RDF/XML. Por exemplo, o grafo da Figura 4 seria representado em RDF da seguinte maneira:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#">
  <contact:Person rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
    <contact:mailbox rdf:resource="mailto:em@w3.org"/>
    <contact:personalTitle>Dr.</contact:personalTitle>
  </contact:Person>
</rdf:RDF>
```

Contudo, os usuários da RDF necessitam também de habilidades para definir vocabulários a serem utilizados em tais declarações, especificamente, para indicar que se está descrevendo tipos ou classes de recursos específicos e que se utilizarão propriedades específicas na descrição destes recursos. Tais classes e propriedades são, segundo Brickley e Guha (2004), descritas através do Vocabulário de Linguagem de Descrição RDF, ou **RDF Schema**. Por exemplo, na definição de classes podem-se utilizar as tags: `rdfs:Resource`, `rdfs:Class`, `rdfs:Literal`, `rdfs:Datatype`, `rdf:XMLLiteral` e `rdf:Property` enquanto que na definição de propriedades podem-se utilizar: `rdfs:range`, `rdfs:domain`, `rdf:type`, `rdfs:subClassOf`, `rdfs:subPropertyOf`, `rdfs:label` e `rdfs:comment`; entre várias outras. Através desta nova camada da *framework* podem-se definir classes – as quais podem ter múltiplas subclasses e superclasses – e propriedades – que podem possuir sub-propriedades, domínio e faixa – elementos estes que a configuram como uma linguagem ontológica simples.

Porém, a fim de alcançar a inter-operabilidade entre vários esquemas diferentes, semânticas mais ricas são necessárias. Por exemplo, o RDF Schema não possibilita a especificação de que as classes *Caminhão* e *Carro* são disjuntas, ou de que um carro possui exatamente quatro rodas. Assim surge a necessidade de evolução para uma linguagem ontológica completa, visto que, segundo Heflin (2004), ontologias são críticas para aplicações em que se deseja a busca ou a reunião de informações originárias de diversas origens.

2.4.3.2 Linguagens de Representação de Ontologias na Web

Na especificação da OWL Web Ontology Language, a W3C aproveita-se da habilidade inerente ao padrão XML em definir esquemas de tags customizadas, da abordagem flexível do padrão RDF para representar dados e das bases semânticas do padrão RDF Schema. A OWL possibilita uma maior expressividade sobre o conteúdo da Web – do que aquela oferecida por XML, RDF e RDF Schema – ao prover um vocabulário adicional juntamente com uma semântica formal. OWL tem, segundo McGuinness e Harmelen (2004), três sub-linguagens de expressividade incremental:

OWL *Lite* se presta a usuários que necessitam essencialmente de uma classificação hierárquica e restrições simples. Por exemplo, embora suporte restrições de cardinalidade, ela só permite valores de cardinalidade zero ou um. OWL *Lite* oferece um caminho de migração mais rápido de tesouros e outras taxonomias. OWL *Lite* também tem uma menor complexidade formal do que OWL DL;

OWL DL visa aos usuários que desejam a máxima expressividade mantendo completude computacional (todas as consequências lógicas são garantidas de serem computáveis) e decidibilidade (todas as computações terminarão em tempo finito). OWL DL inclui todas as construções da linguagem OWL, mas elas podem ser usadas somente sob certas restrições (por exemplo, enquanto uma classe pode ser subclasse de muitas classes, ela não pode ser uma instância de outra classe). OWL DL é assim chamada devido ao seu embasamento na Lógica de Descrição;

OWL *Full* é direcionada para usuários que querem a máxima expressividade e a liberdade sintática do RDF mas, sem nenhuma garantia computacional. Por exemplo, em OWL *Full* uma classe pode ser tratada simultaneamente como uma coleção de indivíduos e como um simples indivíduo. OWL *Full* permite que uma ontologia amplie o significado pré-definido de vocabulários (RDF e OWL). É improvável que qualquer software de raciocínio será capaz de suportar completamente toda a expressividade da OWL *Full*.

Assim, a OWL foi projetada para o uso em aplicações que necessitam processar o significado de conteúdos expressos por meio de ontologias. Por exemplo, tome-se a expressão DL a seguir, a qual define o conceito PapelJurisprudencia:

$$\text{PapelJurisprudencia} \equiv \exists \text{ehPapelJurisprudenciaDe.Ator} \\ \sqcap (\text{Presidente} \sqcup \text{Recorrente} \sqcup \text{Recorrido} \sqcup \text{Relator} \sqcup \text{Revisor})$$

Convertendo a expressão acima para a sintaxe da OWL-DL, tem-se:

```

1: <owl:Class rdf:about="#PapelJurisprudencia">
2:   <owl:equivalentClass>
3:     <owl:Class>
4:       <owl:intersectionOf rdf:parseType="Collection">
5:         <owl:Restriction>
6:           <owl:someValuesFrom rdf:resource="#Ator"/>
7:           <owl:onProperty>
8:             <owl:ObjectProperty rdf:ID="ehPapelJurisprudenciaDe"/>
9:           </owl:onProperty>
10:        </owl:Restriction>
11:       <owl:Class>
12:         <owl:unionOf rdf:parseType="Collection">
13:           <owl:Class rdf:about="#Presidente"/>
14:           <owl:Class rdf:about="#Recorrente"/>
15:           <owl:Class rdf:about="#Recorrido"/>
16:           <owl:Class rdf:about="#Relator"/>
17:           <owl:Class rdf:about="#Revisor"/>
18:         </owl:unionOf>
19:       </owl:Class>
20:     </owl:intersectionOf>
21:   </owl:Class>
22: </owl:equivalentClass>
23: </owl:Class>

```

onde:

os numerais à esquerda, seguidos de ‘:’, não pertencem à sintaxe da OWL, figurando no exemplo apenas como numeradores de cada linha do código;

owl:Class é uma tag que abre uma seção de declarações de inclusão ou definição de uma classe;

owl:equivalentClass é uma tag que abre uma seção de definições da classe correntemente declarada;

owl:intersectionOf é uma tag que abre uma seção de intersecção entre classes, afirmações ou axiomas que definem a classe correntemente declarada;

owl:Restriction é uma tag que abre uma seção de restrição à classe correntemente declarada em relação a outra classe, através de uma propriedade relacional entre elas;

owl:someValuesFrom é uma tag que especifica que o tipo de restrição é existencial, declarando qual é a classe relacionada;

owl:onProperty é uma tag que abre uma seção de declaração da propriedade envolvida na corrente afirmação;

owl:ObjectProperty é uma tag que especifica uma propriedade de objeto (ou relacional) envolvida na corrente afirmação;

owl:unionOf é uma tag que abre uma seção de união entre classes, afirmações ou axiomas que definem a classe correntemente declarada;

linhas de 5 a 10 efetuam a definição da afirmação explícita “ \exists ehPapelJurisprudenciaDe.Ator”;

linhas de 11 a 19 efetuam a definição do axioma de cobertura¹⁰

“(Presidente \sqcup Recorrente \sqcup Recorrido \sqcup Relator \sqcup Revisor)”;

linhas de 4 a 20 efetuam a intersecção entre a afirmação e o axioma descritos nos dois itens anteriores.

2.4.3.3 Linguagem de Consultas em Ontologias na Web

RDF, RDF Schema e OWL são recomendações da W3C para representação e armazenagem de conhecimento, sendo cada um baseado no seu antecessor adicionado de um maior nível de expressividade. Em consequência disso surge uma importante questão: **como buscar e manipular dados em grafos RDF ou OWL?** Em resposta a esta questão a W3C definiu a *Protocol And RDF Query Language* (SPARQL), conforme Prud’Hommeaux e Seaborne (2008), tendo sido construída, segundo McCarthy (2005), a partir de linguagens anteriores de consultas em RDF, tais como *rdfDB*, *RDQL* e *SeRQL*, adicionando novas funcionalidades.

Como uma linguagem de consulta SPARQL é, segundo Hewlett-Packard (2010), “orientado a dados” posto que os resultados baseiam-se exclusivamente nas informações contidas nos modelos – não há inferência. SPARQL não faz outra coisa senão tomar a descrição do que a aplicação deseja, sob a forma de uma consulta, e retornar a informação, sob a forma de um conjunto de ligações ou um grafo RDF. Tome-se, como exemplo, o modelo representado pelo grafo RDF apresentado na Figura 5.

¹⁰Um Axioma de Cobertura obriga que indivíduos de um conceito sejam obrigatoriamente indivíduos de um de seus subconceitos.

Fonte: (MCCARTHY, 2005)

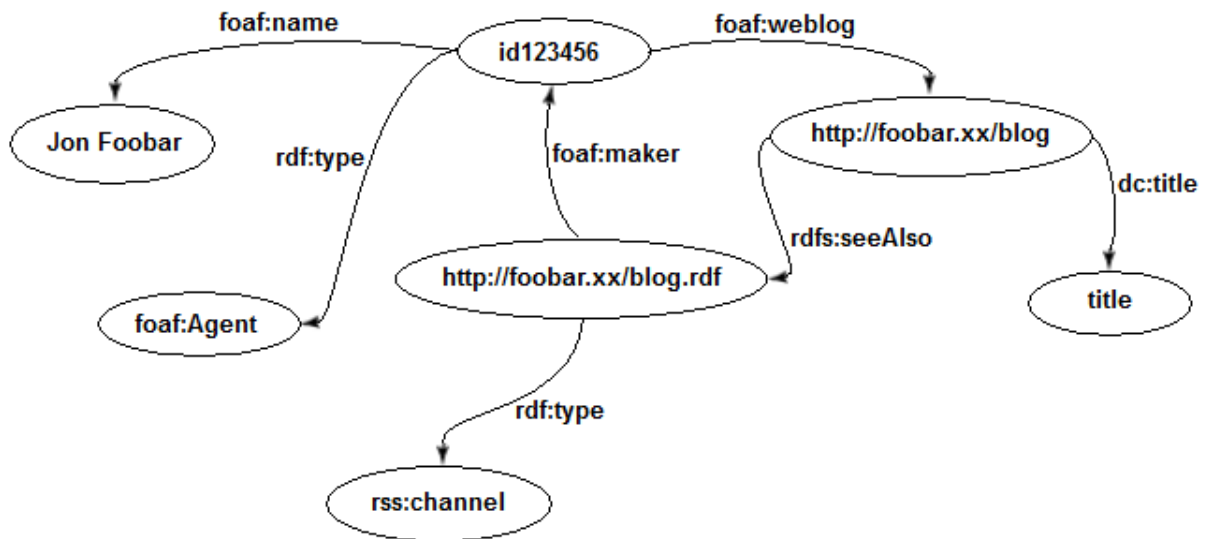


Figura 5: Grafo RDF

Considerando uma consulta “achar a URL do blog pelo nome de pessoa Jon Foobar”, teria-se a seguinte consulta SPARQL:

```

PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?url
FROM <bloggers.rdf>
WHERE {
    ?colaborador foaf:name "Jon Foobar" .
    ?colaborador foaf:weblog ?url .
}
  
```

onde, segundo McCarthy (2005):

- a primeira linha define um prefixo para o *namespace* FOAF¹¹, de maneira que não seja necessária a inclusão da sua URL completa toda vez que for referenciado;
- a cláusula SELECT especifica o que a consulta deve retornar – no caso a variável “url”. Em SPARQL as variáveis são sempre prefixadas ou com ‘?’ ou com ‘\$’;
- FROM é uma cláusula opcional que provê uma URI do *dataset* a ser utilizado. No caso, está se apontando para um arquivo local, porém seria possível se indicar a URL de um grafo existente na Web;

¹¹O projeto *The Friend of a Friend*, FOAF, está criando uma teia de páginas legíveis por máquina descrevendo pessoas, as ligações entre elas e as coisas que elas criam e fazem (BRICKLEY; MILLER, 2000).

- WHERE é uma cláusula que consiste em uma série de *triplas padrão*, as quais, juntas, compõem um *grafo padrão*.

Assim, quando um interpretador executa a consulta, este procura por triplas no modelo de entrada que correspondam às triplas do grafo padrão. Cada variável do grafo padrão que corresponder a um nó do modelo torna-se uma solução à consulta, enquanto que os valores de variáveis presentes na cláusula SELECT integram o resultado da consulta. Neste exemplo, a primeira tripla padrão inserida na cláusula WHERE busca um sujeito que aponte, através de um predicado *foaf:name*, para um objeto literal “Jon Foobar” vinculando-o à variável *colaborador*. No modelo, *colaborador* se equivalerá ao nó em branco do tipo *foaf:Agent*, localizado no topo do grafo. A segunda tripla padrão procura um predicado *foaf:weblog* que parte do sujeito encontrado na primeira tripla (estando vinculado à variável *colaborador*). O objeto apontado por este predicado é associado à variável *url*, tornando-se a solução da consulta, correspondendo, no exemplo, a *http://foobar.xx/blog*.

Nesta subseção foi apresentada uma introdução ao protocolo SPARQL, não fazendo parte do escopo do presente trabalho uma apresentação mais avançada. Entretanto pode-se aqui citar alguns recursos que tornam a linguagem de consultas do SPARQL uma ferramenta de grande potencial para consultas complexas em modelos RDF ou ontologias OWL. Por exemplo: os resultados podem ser refinados através das palavras-chave DISTINCT, LIMIT, OFFSET e ORDER BY, as quais tem funcionamento similar ao SQL; OPTIONAL possibilita a busca de triplas alternativas, tendo um efeito similar ao OR do SQL; UNION possibilita a união de triplas resultantes, através da especificação de múltiplos grafos padrão em uma mesma consulta; FILTER possibilita que se especifique o valor desejado, ou parte dele, dos literais encontrados, restringindo assim o resultado da consulta.

2.4.3.4 Desenvolvimento de Aplicações Web Baseadas em Ontologia

OWL e SPARQL são recomendações especificadas pela autoridade padronizadora da Internet, W3C, visando à criação de aplicações para a Web Semântica baseadas em ontologias. Entretanto a operacionalização de tais padrões em sistemas de informação requer implementação, sendo um exemplo a **Jena**, uma framework de código aberto implementada em Java para construção de aplicações de Web Semântica que, segundo Hewlett-Packard (2010), possibilita a gestão de conhecimento em ontologias, representadas na linguagem OWL (além de RDF e RDF Schema), através de uma Application Programming Interface (API) e de um “motor” para consultas SPARQL.

A API *Jena RDF* pode ser utilizada, segundo Hewlett-Packard (2010), para criar e manipular grafos RDF, através de classes que representam grafos, sujeitos, predicados e objetos literais, denominados na interface da Jena, respectivamente, de modelos, recursos, propriedades e literais. Os grafos criados podem, ao final, ser impressos ou gravados em arquivo.

Além destas funcionalidades iniciais, podem-se citar também: a carga e instanciação em memória de grafos pré-existentes em arquivos ou URLs em sintaxe RDF/XML; a prefixação de namespaces; navegação e busca (limitada) em modelos; união, intersecção e subtração entre modelos inteiros; criação de indivíduos correlatos a recursos existentes no modelo; adição de propriedades e literais a indivíduos; entre outras. De fato, a API Jena RDF é bastante extensa, possibilitando uma completa modelagem de T-Box, bem como a instanciação da respectiva A-Box incluindo muitos milhares de indivíduos. Além disso, a framework Jena possibilita consultas avançadas em grafos ou ontologias através do ARQ, um interpretador SPARQL que implementa o protocolo especificado pela W3C.

2.4.4 Metodologias para desenvolvimento e gestão de ontologias

São várias as razões para se desenvolver uma ontologia, segundo Noy e McGuinness (2000): a) compartilhar o entendimento comum da estrutura de informação entre pessoas ou sistemas; b) permitir a reutilização do conhecimento do domínio; c) fazer pressupostos explícitos de domínio; d) separar o conhecimento de domínio do conhecimento operacional; e) analisar o conhecimento de domínio. Entretanto, toda solução que sistematiza o conhecimento, requer que o mesmo seja, segundo York, Steffen e Studer (2003), modelado, apropriadamente estruturado e interligado para o devido suporte à sua apresentação e gestão. Existem várias propostas de metodologias para a modelagem e a gestão do conhecimento em empresas, as quais, geralmente são divididas em duas fases:

Metaprocesso de Conhecimento, responsável por desenvolver e manter uma solução de conhecimento em uma organização;

Processo de Conhecimento, o qual visa à utilização da solução criada na primeira fase.

Como exemplo, podem-se citar as seguintes metodologias ou abordagens, tidas como as mais conhecidas ou difundidas: CommonKADS; DOGMA; Enterprise Ontology; KACTUS; Methontology; On-To-Knowledge Methodology (OTKM); SENSUS; TOVE.

A escolha de uma metodologia deve levar em consideração a sua adequação à natureza e extensão do problema. Para uma análise mais detalhada pode-se utilizar uma metodologia de

comparação por quesitos de gerenciamento de projeto, pré-desenvolvimento, desenvolvimento, pós-desenvolvimento, integração, avaliação e documentação como proposto por Silva, Souza e Almeida (2008), onde pode-se, inclusive, consultar um quadro comparativo entre algumas metodologias para auxiliar na decisão.

Das metodologias acima pode-se destacar a Methontology, definida por Fernández, Gómez-Pérez e Juristo (1997), a qual possui um alto nível de detalhamento e documentação para a fase de Metaprocessos de Conhecimento, sendo, assim, bastante adequada para grandes domínios de conhecimento, tais como: elementos químicos, componentes poluentes ambientais e páginas amarelas (GÓMEZ-PÉREZ, 1999).

Outra metodologia bastante difundida é a OTKM, definida por York, Steffen e Studer (2003), a qual apresenta formalismo e extensão medianos, sendo assim adequada para problemas envolvendo domínios de até médio porte. Um diferencial da OTKM, com relação à Methontology, é a existência da fase Processo de Conhecimento, dando, assim, cobertura para o ciclo de vida completo de sistemas de gestão do conhecimento. Estas características tornam a OTKM uma opção viável para o presente trabalho, posto que envolve um domínio de escopo reduzido e, além disso, necessita do desenvolvimento de uma solução para utilização do conhecimento.

2.4.5 On-To-Knowledge Methodology (OTKM)

Criada à partir de um projeto europeu de mesmo nome, liderado por York Sure, Steffen Staab e Rudi Studer, a OTKM (YORK; STEFFEN; STUDER, 2003) é constituída pelos dois tradicionais processos do ciclo de desenvolvimento e gestão do conhecimento, como citadas na subseção anterior.

2.4.5.1 Metaprocessos de Conhecimento

O Metaprocessos de Conhecimento é composto de 5 fases, entre os quais há sempre uma decisão a ser tomada e resultados específicos a serem gerados, como representado na Figura 6.

2.4.5.2 Fase Estudo de Viabilidade

Muitos fatores, independentemente de tecnologia, podem determinar o sucesso ou o fracasso de um sistema. Assim, a primeira fase do metaprocessos, pretende analisar tais fatores através de um estudo de viabilidade, visando a identificar problemas ou oportunidades e potenciais soluções (YORK; STEFFEN; STUDER, 2003). Para auxiliar esta tarefa, pode-se elaborar

Fonte: (YORK; STEFFEN; STUDER, 2003) – tradução livre do autor

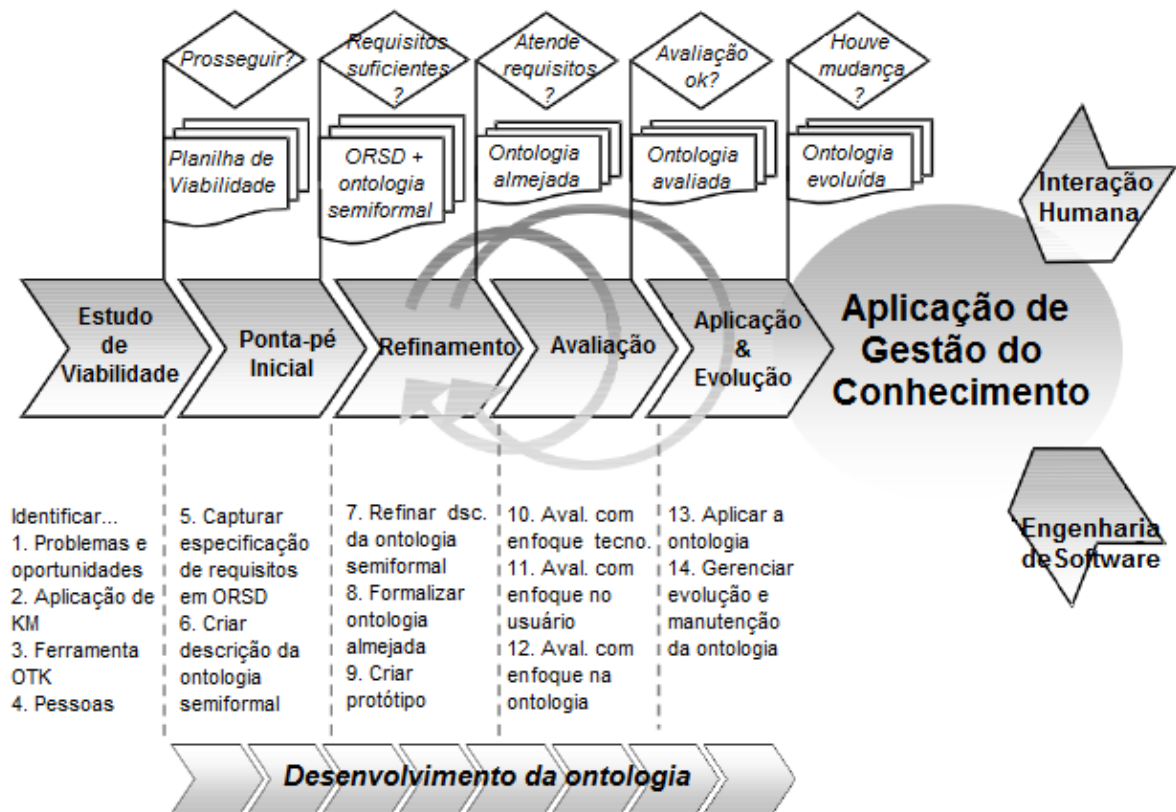


Figura 6: O Metaprocesso de Conhecimento

uma planilha com as seguintes informações:

Problemas e Oportunidades, que corresponde ao passo 1, conforme a Figura 6, e visa a identificar os problemas que deverão ser abordados na realização do projeto. A ideia não é simplesmente inserir cada fase padrão da OTKM, mas identificar as atividades macro, sub-projetos, módulos ou iterações que podem quebrar a complexidade do projeto, variando, naturalmente, a cada domínio ou aplicação;

Aplicação de KM (Knowledge Management), que corresponde ao passo 2 e pretende listar as tarefas de desenvolvimento ou gestão de conhecimento necessárias para solucionar cada *problema ou oportunidade* identificado para o projeto;

Ferramenta OTK, que corresponde ao passo 3 e lista quais passos ou fases da OTKM serão utilizados para se desenvolver a *Aplicação de KM* identificada;

Pessoas, corresponde ao passo 4 e visa a identificar as pessoas envolvidas na execução da *ferramenta OTK* identificada.

Na OTKM toda fase termina com uma decisão a ser tomada, baseada nos artefatos gerados ao longo da mesma e em uma pergunta pré-estabelecida. Para o Estudo de Viabilidade a pergunta é: **PROSEGUIR?** A resposta é dada pelo projetista, baseado na razão, não havendo uma metodologia para orientar a sua obtenção. Entretanto, o bom senso pode levar a uma resposta positiva caso a maioria dos problemas levantados demonstrem uma viabilidade média ou alta.

2.4.5.3 Fase Ponta-pé Inicial

Nesta fase, o desenvolvimento da ontologia efetivamente se inicia, através de dois artefatos (YORK; STEFFEN; STUDER, 2003):

Documento de Especificação de Requisitos da Ontologia (ORSO), o qual descreve o que a ontologia deve cobrir, a aplicação da ontologia e fontes de conhecimento que podem ser utilizadas no desenvolvimento do próximo artefato;

Descrição semiformal da ontologia, através de um grafo composto dos seguintes elementos: nós nomeados (representando conceitos), arcos nomeados ou não (representando o relacionamento entre os conceitos) e textos descritivos, quando desejado, ligados aos nós ou aos arcos, semelhantemente a um mapa mental (BUZAN, 2000).

Ao final desta fase a pergunta a ser respondida é: **REQUISITOS SUFICIENTES?** A resposta deve ser obtida avaliando se os artefatos obtidos, ORSO e grafo, dispõem de informações suficientes para o desenvolvimento da ontologia.

2.4.5.4 Fase Refinamento

O objetivo desta fase é a formalização da ontologia semiformal, resultante da fase anterior, até a obtenção da ontologia almejada, através dos seguintes passos:

Refinamento da ontologia semiformal: neste passo, correspondente ao 7 da Figura 6, deve-se formar uma taxonomia com os conceitos, levantados na descrição semiformal, adicionando outros tipos de relações, além das *is-a*, inerentes aos modelos taxonômicos. Além disso, com a ajuda de especialistas nos domínios, devem-se adicionar atributos aos conceitos e traçar novas relações entre os conceitos.

Formalização da ontologia almejada e criação de um protótipo: neste passo, correspondente ao 8 e ao 9, a taxonomia resultante da fase de refinamento, deve já conter os principais

conceitos e propriedades, tanto de dados como de relacionamento, suficientes para construir a primeira versão da ontologia almejada, bem como o respectivo protótipo, visando a se efetuar testes de consistência. Neste ponto, é aconselhável a seleção de ferramentas que auxiliem na edição de ontologias e na prototipação. Ambas tarefas podem ser efetuadas, por exemplo, através do Protégé (INFORMATICS, 2010), ferramenta criada e mantida pela Universidade de Stanford. Através deste sistema pode-se modelar e criar a ontologia, através da linguagem OWL. O Protégé possibilita também a prototipação, disponibilizando interface para inserção de instâncias de conceitos e de valores de propriedades, a qual efetua consistência semântica, durante a inserção, bem como o reconhecimento de conceitos que definem instâncias, após a sua inserção.

O confronto da primeira versão da ontologia com o ORSD deve ser suficiente para responder à questão a ser decidida ao final desta fase: **A ONTOLOGIA ATENDE AOS REQUISITOS LEVANTADOS?**

2.4.5.5 Fase Avaliação

A ontologia construída deve ter garantia de qualidade, o que, segundo York, Steffen e Studer (2003), pode ser aferida através de três tipos de avaliação, conforme os passos de 10 a 12 da Figura 6:

Avaliação com enfoque tecnológico: deve-se avaliar a ontologia, através do uso de ferramenta que avalie automaticamente propriedades (sintaxe), consistência (semântica) e propriedades tecnológicas (e.g. inter-operabilidade, escalabilidade, etc). O Pellet é um exemplo de opção para avaliação sintática e semântica, consistindo em um programa “raciocinador” que pode ser invocado a partir da interface do Protégé visando a submeter a ontologia presente na memória. Segundo Parsia (2010), o Pellet suporta os seguintes construtores da linguagem OWL:

- restrições qualificadas de cardinalidade;
- axiomas complexos de subpropriedades;
- restrições locais de reflexividade;
- propriedades reflexivas, não-reflexivas, simétricas e assimétricas;
- propriedades de disjunção;
- dados definidos pelo usuário.

Para avaliar uma ontologia o Pellet, ainda segundo Parsia (2010), realiza as seguintes tarefas de raciocínio:

Avaliação de Consistência: visa a garantir que a ontologia não possua contradições, baseado nas definições de consistência de linguagem de ontologia, conforme ditadas pela W3C (MOTIK et al., 2009);

Satisfabilidade de Conceitos: determina se é possível uma classe possuir instâncias, através do cálculo de satisfabilidade, de forma que, se uma classe for insatisfazível, a ontologia toda será considerada inconsistente;

Classificação: gera, sempre que possível, relações de subclasse entre todas as classes da ontologia (desde que ainda não hajam tais relações) visando a criar uma hierarquia completa e adequada a responder consultas de subclasses diretas ou indiretas;

Realização: determina a classe a que pertence um indivíduo. Esta tarefa depende que uma hierarquia completa de classes já tenha sido definida pela tarefa de Classificação;

Avaliação com enfoque no usuário: o objetivo aqui, segundo York, Steffen e Studer (2003), é avaliar a satisfação do usuário com a aplicação de Gerenciamento de Conhecimento e se a aplicação, baseada em ontologia, é no mínimo tão boa quanto as aplicações existentes para resolver problemas similares.

Avaliação com enfoque na ontologia: York, Steffen e Studer (2003) determina que é necessário que **alguém** avalie formalmente a ontologia criada, sendo que a abordagem mais recomendada é a OntoClean (GUARINO; WELTY, 2002), a qual, baseada em noções filosóficas, conduz a hierarquias de ontologias mais corretas. Este tipo de avaliação pode provocar o retorno à fase de refinamento, visando à criação de uma nova versão da ontologia, baseado no resultado deste passo;

A decisão, ao final desta fase, se baseia na pergunta: **A ONTOLOGIA AVALIADA PREENCHE TODOS OS CRITÉRIOS DE AVALIAÇÃO RELEVANTES PARA A APLICAÇÃO ALMEJADA?**

2.4.5.6 Fase Aplicação e Evolução

A aplicação de ontologias em sistemas produtivos, segundo York, Steffen e Studer (2003), é descrita da mesma maneira que o Processo de Conhecimento. Como geralmente a fase de aplicação implica na construção de um sistema aplicativo, ela poderá demandar, adicionalmente, da utilização de uma metodologia de desenvolvimento de *software*.

A evolução deve consistir em um processo organizacional regido por regras de atualizações, inclusões ou exclusões, sendo recomendável o controle de mudanças através de versionamento.

A Figura 6, página 68, que resume graficamente o Metaprocesso, demonstra que os passos de Refinamento, Avaliação e Aplicação são cíclicos, mas, geralmente, o retorno à fase de refinamento é necessário somente para adições de conceitos específicos ou periféricos que passaram despercebidos pela fases iniciais, demonstrando assim uma boa qualidade na execução do processo. Entretanto, dada a natureza que toda aplicação baseada em ontologia deve ter, de independência entre os dados e a arquitetura dos sistemas, podendo chegar até um ponto ideal onde as regras de negócio estão presentes exclusivamente na ontologia e não nos códigos-fonte, mudanças drásticas nos conceitos ou propriedades de base não devem prejudicar o sistema aplicativo da ontologia, sendo uma prática até bem-vinda, se analisada do ponto de vista de evolução da aplicação.

2.4.5.7 Processo de Conhecimento

Uma vez que a aplicação de Gestão de Conhecimento esteja totalmente implantada na organização, o Processo de Conhecimento essencialmente “gira”, segundo York, Steffen e Studer (2003), em torno de quatro passos, dispostos em um ciclo contínuo, que permeiam a devida utilização da aplicação, conforme demonstrados na Figura 7.

Fonte: (YORK; STEFFEN; STUDER, 2003) – tradução livre do autor

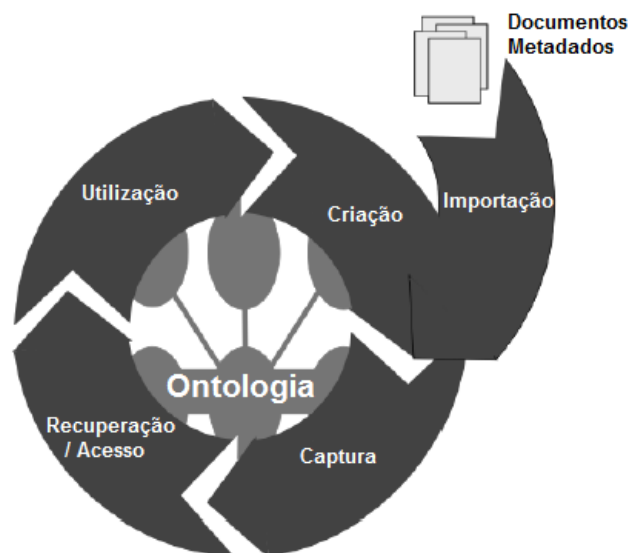


Figura 7: O Processo de Conhecimento

O presente trabalho não cobre a implantação imediata da aplicação no TJPR, visando, tão somente, ao estudo da viabilidade e das vantagens de se utilizar uma aplicação baseada em

ontologia no processo de pesquisa jurisprudencial do referido tribunal. Isto posto, nada mais será acrescentado aqui a respeito desta fase da OTKM, visto que extrapola o escopo proposto.

2.5 AVALIAÇÃO DE RESULTADOS DE PESQUISAS

Existem inúmeras maneiras de se avaliar os resultados de uma pesquisa, sendo as mais comuns os cálculos dos índices de **Precisão** e **Revocação**. Ambas métricas se baseiam nas seguintes premissas, conforme a Figura 8: **a)** é conhecido o conjunto de dados retornado da pesquisa; **b)** é conhecido o conjunto de dados relevantes existente na base de dados.

Fonte: (JIZBA, 2000) – tradução livre do autor

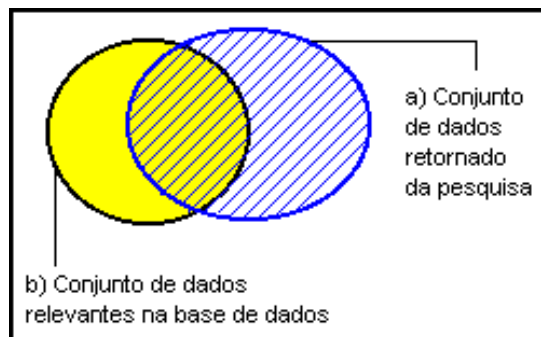


Figura 8: Precisão e Revocação – Premissas

A partir destas premissas, as métricas subentendem três fatos, conforme a Figura 9: **c)** há um conjunto de dados irrelevantes retornado pela pesquisa; **d)** há um conjunto de dados relevantes retornado pela pesquisa; **e)** há um conjunto de dados relevantes que não retornaram na pesquisa.

Fonte: (JIZBA, 2000) – tradução livre do autor

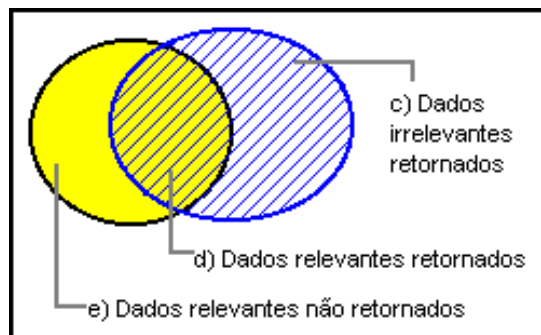


Figura 9: Precisão e Revocação – Fatos

2.5.1 Precisão

Dadas as premissas e os fatos acima, pode-se enunciar a Precisão (P) como o percentual de dados relevantes retornados (d na Figura 9) dentre o total de dados retornados pela pesquisa (a na Figura 8), podendo então ser representada pela equação $P = \frac{d}{a} \cdot 100$ e pela Figura 10.

Fonte: (JIZBA, 2000) – tradução livre do autor

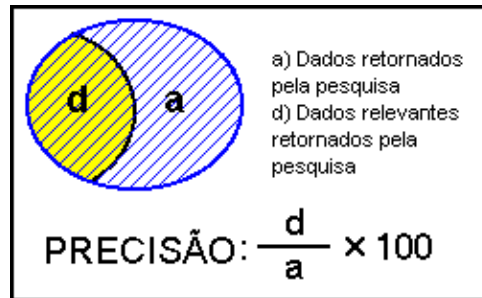


Figura 10: Precisão

Assim, o índice de Precisão demonstra qual foi o percentual de dados relevantes do total retornado pela pesquisa.

2.5.2 Revocação

A Revocação (C), também baseada nas premissas e fatos acima, pode ser enunciada como o percentual de dados relevantes retornados (d na Figura 9) que estão dentre o total de dados relevantes existentes na base de dados (b na Figura 8), podendo então ser representada pela equação $C = \frac{d}{b} \cdot 100$ e visualizada pela Figura 11.

Fonte: (JIZBA, 2000) – tradução livre do autor

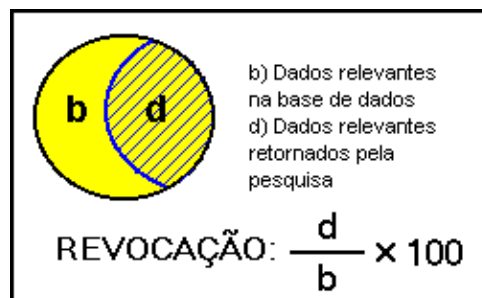


Figura 11: Revocação

Assim, o índice de Revocação demonstra qual foi o total de dados retornados dentre o total de dados relevantes na base de dados.

3 MÉTODOS

Este capítulo descreve os métodos utilizados na criação da ontologia que representa o domínio dos acórdãos da jurisprudência do TJPR, bem como da respectiva construção da aplicação de gestão. Ao longo do capítulo serão apresentados artefatos intermediários correspondentes a cada passo do método, o qual visa à obtenção dos resultados finais, conforme apresentados no capítulo 4.

Direcionando as tarefas e técnicas a seguir descritas, está a metodologia OTKM com diretrizes que guiam todo o processo de desenvolvimento, visando à solução do problema. O artigo (YORK; STEFFEN; STUDER, 2003), o qual introduz a OTKM, não especifica padrões de documentação ou técnicas para criação de artefatos, limitando-se apenas à explicitação das diretrizes que orientam o ciclo de vida da ontologia. Desta forma, os artefatos resultantes, aqui expressos, consistem em um esforço do autor em interpretar (e seguir) as diretrizes preconizadas na metodologia. O diagrama da Figura 12 ilustra (e resume) graficamente os métodos propostos.

Fonte: adaptação de (YORK; STEFFEN; STUDER, 2003)

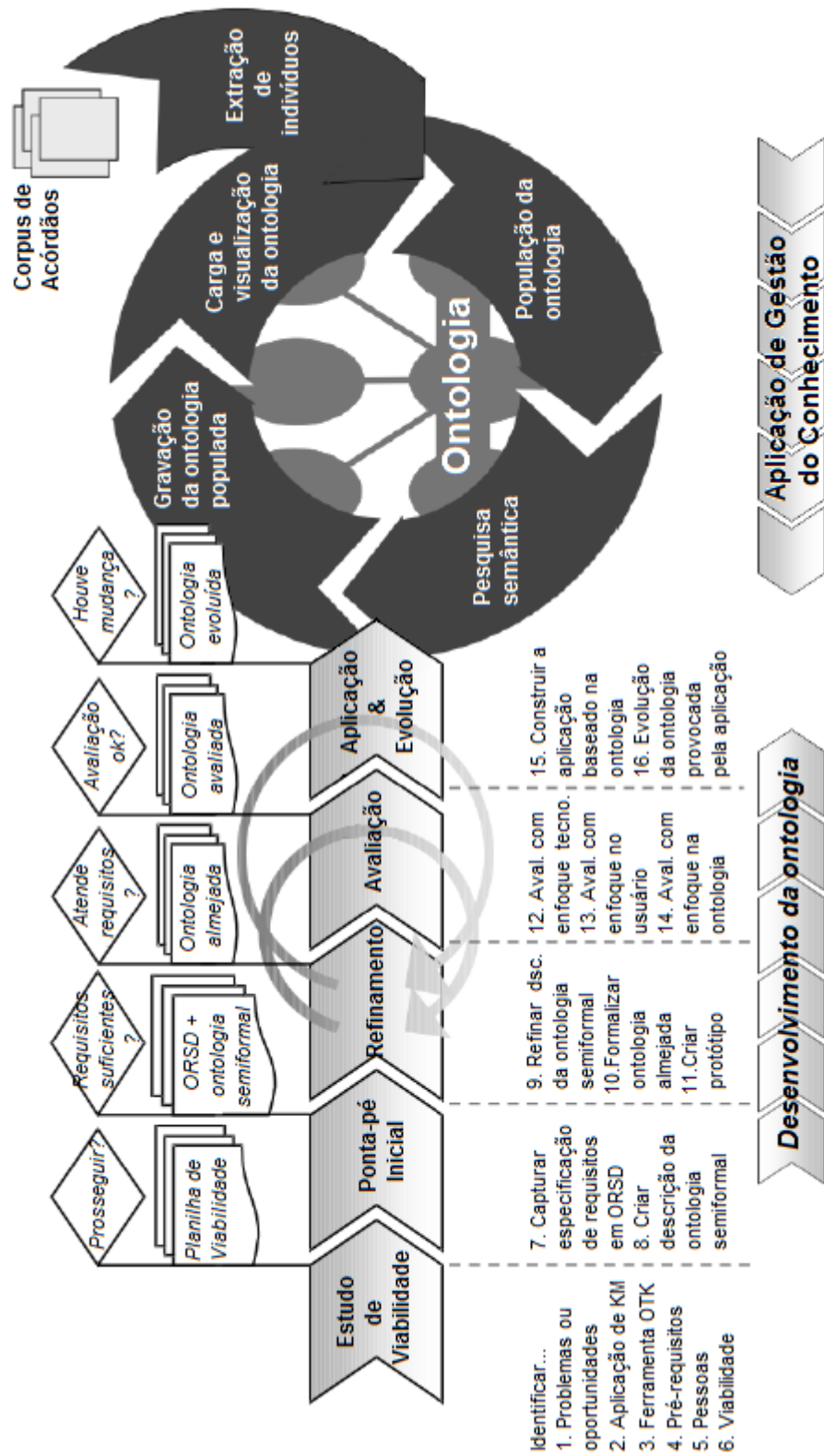


Figura 12: Diagrama métodos propostos

3.1 METAPROCESSO DE CONHECIMENTO

Antes de iniciar o processo de criação da ontologia, foi necessário um estudo visando ao conhecimento do domínio dos acórdãos. Primeiramente estudaram-se vários documentos publicados na jurisprudência do TJPR, como por exemplo aquele contido no anexo A, o que foi favorecido pela experiência do autor de mais de oito anos no desenvolvimento de sistemas de informações de processos judiciais, no mesmo tribunal. Em um segundo momento foram entrevistados especialistas da área de Direito.

3.1.1 Aplicação da Fase Estudo de Viabilidade

Com o conhecimento inicial adquirido, foi possível realizar o **Estudo de Viabilidade**, correspondente à primeira fase da OTKM. Para ampliar a funcionalidade da planilha de viabilidade, foram adicionadas duas novas colunas: **Pré-requisito**, que indica os recursos necessários para operacionalizar a ferramenta OTK ou a tarefa da aplicação em questão; **Viabilidade**, a qual visa a estimar a viabilidade de se realizar a ferramenta OTK ou a tarefa da aplicação. Assim, no Quadro 5 pode-se ver o resultado deste estudo efetuado para o problema aqui estudado.

Problema ou Oportunidade	Aplicação de KM	Ferramenta OTK	Pré-requisitos	Pessoas	Viab.
Estudo e identificação de metodologia	Pré-requisito		Artigos, bibliotecas de funções, orientação	Autor	Alta
Conceitualização	Desenvolvimento de ontologia semi-formal	<i>2ªF.</i> Ponta-pé inicial	Conhecimento do domínio	Autor	Alta
		<i>3ªF.</i> Refinamento – <i>P7.</i> Refinar a descrição semi-formal	Entrevistas com especialistas do Direito	Autor	Média
Construção	Criação da ontologia utilizando	<i>3ªF.</i> Refinamento – <i>P8.</i> Formalizar ontologia	Operacionalização do Protégé	Autor	Alta

(continua)

Quadro 5: Planilha de Viabilidade

Problema ou Oportunidade	Aplicação de KM	Ferramenta OTK	Pré-requisitos	Pessoas	Viab.
	OWL-DL	<i>3ª F.</i> Refinamento – <i>P9.</i> Criar protótipo			
Avaliação	Consistência da ontologia	<i>4ª F.</i> Avaliação – <i>P12.</i> com enfoque na ontologia	Operacionalização do Pellet	Autor	Alta
	Conformidade da ontologia	<i>4ª F.</i> Avaliação – <i>P11.</i> com enfoque no usuário	Cooptar especialistas de jurisprudência	Seção Jurisp. TJPR	Baixa
Aplicação	Composição de corpus		Acesso a banco de dados de jurisprudência	Autor	Média
	Processamento de informações do corpus	<i>5ª F.</i> Aplicação – <i>P13.</i> Aplicar a ontologia	Criar programa de extração de conceitos jurisprudenciais	Autor	Alta
	População da ontologia		Criar módulo de população da ontologia	Autor	Alta
	Pesquisa na ontologia		Criar módulo de pesquisa na ontologia	Autor	Alta
	Testes da extração	<i>4ª F.</i> Avaliação – <i>P10.</i> com enfoque na tecnologia	Descrever método de avaliação	Autor	Alta
	Testes da pesquisas		Descrever método de avaliação	Autor	Alta

Quadro 5: Planilha de Viabilidade

Onde:

Estudo e identificação de metodologia adequada trata-se de um pré-requisito para o início das aplicações de KM, sendo que deve-se efetuar um estudo inicial das atuais metodologias existentes para a criação de ontologias visando a selecionar a mais adequada para a aplicação corrente (neste caso foi selecionada a própria OTKM). Dada a sabida disponibilidade que há de estudos científicos do assunto bem como de propostas metodológicas, a viabilidade de solução para este problema foi considerada alta;

Conceitualização é o primeiro desafio para viabilizar o projeto, consistindo na organização mental das entidades, relações e instâncias do domínio em questão. Este problema é abordado através da aplicação de KM de **Desenvolvimento de ontologia semi-formal** a qual se compõem pelas seguintes ferramentas OTK:

- 2ª Fase – Ponta-pé inicial: composta pelos passos 7 e 8, visa a obter a primeira versão da ontologia semi-formal. Esta fase tem como pré-requisito um conhecimento suficiente do domínio em questão, podendo ser executada pelo próprio autor, cuja experiência em sistemas de domínio de processos judiciais torna a viabilidade alta;
- 3ª Fase – Refinamento: através do passo 9, visa a refinar a descrição semi-formal da ontologia obtida na fase anterior, para tal tem-se um pré-requisito de se efetuar entrevistas com especialistas do Direito, visando à validação daquela primeira versão, tarefa esta que tem viabilidade média, devido ao acesso que o autor tem a profissionais do Direito, mas que, entretanto, tem disponibilidade restrita;

Construção é a atividade em que ocorre a construção da ontologia, descrita através da atividade anterior, sendo que aqui se objetiva o uso da linguagem OWL-DL para tal. Assim, a construção se dá através dos passos 10 e 11, ainda da 3ª fase, nas quais se formaliza a ontologia almejada culminando com a construção de um protótipo. A realização destes passos têm como pré-requisito a operacionalização do Protégé, ou seja a obtenção, instalação e instrução de uso desta ferramenta. Dada a disponibilidade de pacote de instalação e ampla documentação na Internet do produto, o qual é licenciado através das regras de Software Livre, a viabilidade desta atividade é considerada alta;

Avaliação é a atividade onde se verifica a qualidade da ontologia, construída durante a atividade anterior, através de duas tarefas:

- Consistência, que visa a conferir, através de um programa raciocinador, se o relacionamento entre os conceitos estão consistentes, tarefa esta realizada conforme especificado no passo 14 (Avaliação com enfoque em Ontologia) da 4ª fase (Avaliação),

cujo pré-requisito é a operacionalização do Pellet. A viabilidade desta tarefa é alta, considerando as condições de instalação e documentação semelhantes às do Protégé;

- Conformidade, que visa a conferir, através de revisão por especialistas, se os conceitos e propriedades da ontologia estão de acordo com o domínio. Esta tarefa é orientada pelo passo 13 (Avaliação com enfoque no Usuário) da 4ª fase, tem como pré-requisito a cooperação de especialistas em jurisprudência, e por este fato tem uma viabilidade considerada baixa, visto que, apesar da disposição da Seção de Jurisprudência do TJPR, os seus recursos humanos são escassos;

Aplicação é a atividade de construção e testes da aplicação baseada na ontologia, através das seguintes tarefas:

- Composição de corpus, que visa a compor um conjunto de documentos jurisprudenciais de extensão adequada. O pré-requisito desta tarefa é o acesso ao banco de dados de jurisprudência do TJPR, o que torna a sua viabilidade média, posto que, por se tratarem de informações judiciais reais, podem haver empecilhos institucionais;
- Processamento de informações do corpus, que visa à extração de indivíduos do corpus que se adequem aos conceitos da ontologia. Esta tarefa, assim como as duas próximas, se insere no contexto do passo 15 (Construir a aplicação baseado na ontologia) da 5ª fase (Aplicação), tendo como pré-requisito a criação de um programa que efetue todos os processamentos necessários para a extração de informações em cada documento de conceitos jurisprudenciais, o que é de alta viabilidade dada a grande disponibilidade de tecnologias e pesquisas existentes na área de RI;
- População da ontologia, que visa a inserir na ontologia os indivíduos extraídos do corpus, tarefa esta que tem como pré-requisito a criação de módulo do programa que popule a ontologia com o resultado da extração realizada na tarefa anterior, o que pode ser considerado de alta viabilidade devido às definições da W3C da OWL e da disponibilidade da biblioteca de funções Jena para manipulação de ontologias em OWL;
- Pesquisa na ontologia, que visa a disponibilizar uma interface para o usuário efetuar consultas na ontologia populada, tarefa esta que tem como pré-requisito a criação de um módulo que possibilite a submissão de consultas aos indivíduos da ontologia, o que pode ser considerado de alta viabilidade devido às definições da W3C da SPARQL e da disponibilidade de funções para submissão de consultas baseadas nesta linguagem na biblioteca de funções Jena;

- Testes da extração, que visa à avaliação de revocação e precisão dos indivíduos extraídos, correspondendo ao passo 14 (Avaliação com enfoque na tecnologia) da 4ª fase (Avaliação);
- Testes da pesquisa, que visa à avaliação de revocação e precisão do resultado da pesquisa na ontologia populada, correspondendo ao mesmo passo citado na alínea anterior.

Por fim, para a pergunta a ser respondida ao final da fase de estudo de viabilidade, “PROS-Seguir?”, considerou-se o fato de as viabilidades dos problemas terem sido avaliadas com conceitos de média a alta na maioria dos casos, sendo que o único quesito com viabilidade baixa não chega a impedir a conclusão do projeto, levando, assim, à resposta **SIM**.

3.1.2 Aplicação da Fase Ponta-pé inicial

Prosseguindo, na fase do Ponta-pé inicial, baseando-se, ainda, nos conhecimentos iniciais do domínio adquiridos, criou-se o **Documento de Especificação de Requisitos da Ontologia**, conforme o Quadro 6.

Tema	Requisitos
Cobertura da ontologia	Conceitos jurídicos envolvidos com documentos de jurisprudência, tais como: <ul style="list-style-type: none"> - legislação; - órgão julgador; - local do processo; - atores do processo e da jurisprudência; - recurso; - decisão ao recurso; - tipo da jurisprudência.
Aplicação da ontologia	Compartilhamento do conhecimento jurisprudencial entre tribunais. Denotação de semântica a documentos jurisprudenciais semi-estruturados. Fonte de informação para pesquisas semânticas.
Fontes de conhecimento	Documentos jurisprudenciais publicados pelo TJPR. Documentação de sistemas de informações judiciais do TJPR. Tesouro Jurídico da Justiça Federal (CEJ, 1997). Experiência do autor nos citados sistemas. Biblioteca jurídica do TJPR. Especialistas jurídicos do TJPR.

Quadro 6: Documento de Especificação de Requisitos da Ontologia (ORSD)

Assim, é possível o início, de fato, da construção da ontologia, através da **Descrição semi-formal da ontologia**. Baseado na *Cobertura da ontologia* e nas *Fontes de conhecimento*, ambas levantadas no ORSD, construiu-se de um diagrama de classes, conforme a Figura 13.

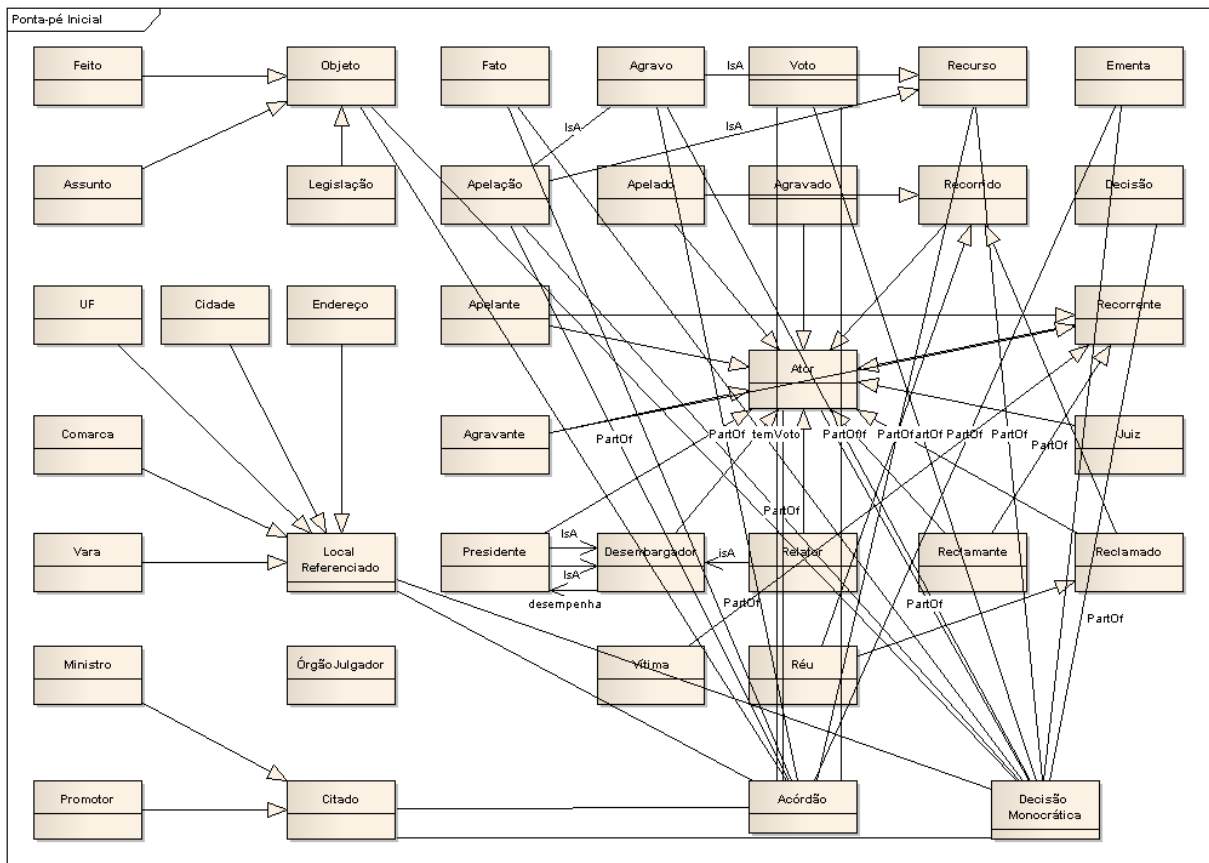


Figura 13: Descrição Semi-formal da Ontologia

Ao final desta fase a pergunta, a ser respondida baseado nos artefatos gerados, é: “REQUISITOS SUFICIENTES?” Em primeira análise, os requisitos levantados e o diagrama de classes inicial são suficientes para prosseguir, logo a decisão é **SIM**.

3.1.3 Aplicação da Fase Refinamento

Nesta fase formalizou-se a ontologia semi-formal, resultante da fase anterior, obtendo-se, assim, uma primeira versão da ontologia almejada, conforme preconizado pela OTKM, através de dois passos.

3.1.3.1 Refinamento da ontologia semi-formal

Neste passo, rearranjou-se o diagrama de classes inicial, formando uma taxonomia com os conceitos levantados, além de se adicionarem atributos a alguns conceitos. Para tal, contou-se com o auxílio de um Advogado, um Assessor Jurídico e uma Assessora de Gabinete de Desembargador, resultando no diagrama da Figura 14.

3.1.3.2 Formalização da ontologia almejada e criação de um protótipo

A ontologia semi-formal refinada contém os principais conceitos e propriedades formulados à partir das fontes de conhecimento enumeradas no ORSD, possibilitando a construção de um protótipo da primeira versão da ontologia almejada utilizando o Protégé, cuja visualização pode-se dar através da Figura 15.



Figura 15: Visualização do protótipo da ontologia JurisTJPR

Maior detalhamento da ontologia, será dado mais adiante, neste trabalho, após sucessivos refinamentos, entretanto, o confronto desta primeira versão com o ORSD é suficiente para responder **SIM** à questão a ser decidida ao final desta fase: “ATENDE AOS REQUISITOS LEVANTADOS?”

3.1.4 Aplicação da Fase Avaliação

Para aferir e garantir a qualidade da ontologia, foram efetuados os três tipos de avaliação propostos na OTKM, porém reordenando a sequência, como aparecem na metodologia, de acordo com a visão deste autor.

3.1.4.1 Avaliação com enfoque na ontologia

Neste quesito adotou-se, como método de avaliação, a apresentação da taxonomia resultante do refinamento da ontologia semi-formal, conforme já apresentado na Figura 14, a uma banca tecnológica composta por 5 doutores e doutorandos integrantes do grupo MEMENTO. Baseado na experiência de cada componente, a banca efetuou uma avaliação dos conceitos, atributos e relações, contidos na taxonomia, visando uma melhor adequação lógica para a ontologia almejada. A opção da taxonomia, como instrumento de avaliação facilitou, a visão hierárquica do domínio daqueles que não estavam diretamente envolvidos com o trabalho.

Assim, esta avaliação provocou um retorno à fase de refinamento (o que é previsto pela OTKM) visando a criação de uma nova versão mais refinada da taxonomia, conforme apresentada na Figura 16, na qual, entre as principais mudanças, sugeridas pela banca, encontram-se:

- remoção do conceito **Decisão Monocrática**, visando uma limitação do escopo deste trabalho, visto que os documentos deste tipo apresentam diferenças na estrutura se comparados aos documentos de Acórdãos;
- reestruturação do conceito de **Ator**, bem como de seus e sub-conceitos, com a criação dos conceitos de **Papéis Jurídicos e Jurisprudenciais**;
- substituição do conceito **Ator** pelos conceitos **Papel, PapelJuridico e PapelJurisprudencia**, sendo estes dois últimos especializações do primeiro;
- os sub-conceitos de **Ator** se tornaram sub-conceitos de **PapelJuridico e PapelJurisprudencia**;
- o conceito **Ator** deixou de ter sub-conceitos e passou a se relacionar com os conceitos **PapelJuridico e PapelJurisprudencia**;
- especialização do conceito **Local Referenciado** em **Físico e Jurídico**, tendo os seus sub-conceitos reagrupados.

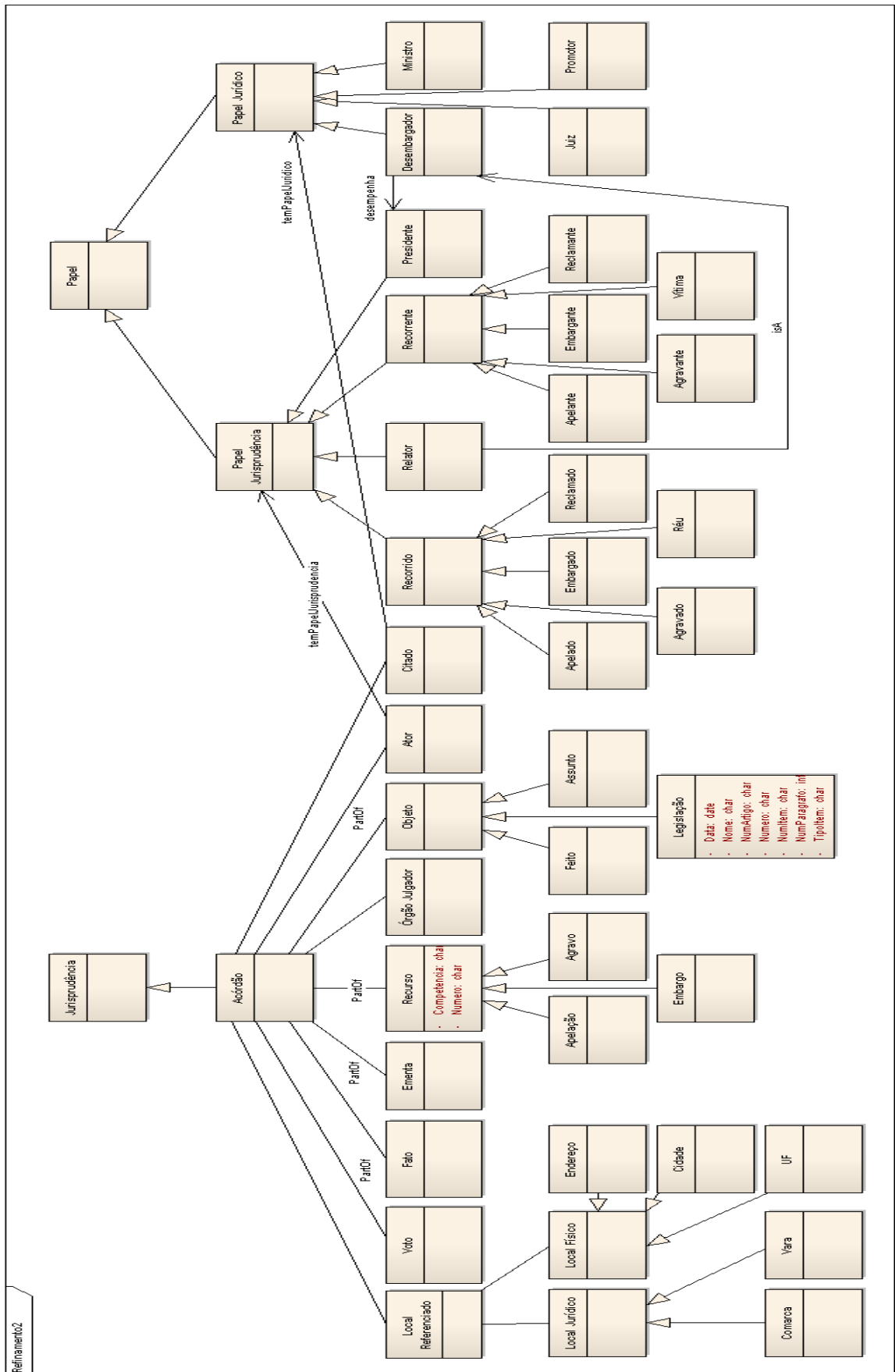


Figura 16: Segundo refinamento da ontologia semi-formal

3.1.4.2 Avaliação com enfoque tecnológico

Conforme preconizado pela OTKM, neste passo a ontologia prototipada no Protégé, foi submetida ao Pellet, enquanto presente na memória daquele aplicativo, visando uma avaliação de consistência. Assim, a submissão da ontologia a este raciocinador resultou em sucesso para todos os seus testes.

3.1.4.3 Avaliação com enfoque no usuário

Para o principal objetivo preconizado pela OTKM para esta avaliação – análise da satisfação do usuário para com a aplicação de gestão do conhecimento – neste ponto não há elementos suficientes, visto que ainda não se chegou à fase de Aplicação, a qual será descrita adiante. A avaliação efetiva da aplicação, com enfoque no usuário, será detalhada na seção 3.3, página 136. Entretanto, dada a óbvia importância da participação dos futuros usuários, buscou-se a participação deles na avaliação da ontologia, visando garantir a sua coerência ao domínio. Assim, a taxonomia resultante do segundo refinamento, conforme a Figura 16, foi apresentada para o pessoal do Centro de Documentação do TJPR, dentro do qual se encontra a Seção de Jurisprudência, integrada por profissionais especializados no assunto correlato, os quais não formularam discordâncias e nem tão pouco acréscimos aos conceitos apresentados.

Por fim, a decisão ao final da fase de Avaliação se baseia na pergunta: “A ONTOLOGIA AVALIADA PREENCHE TODOS OS CRITÉRIOS DE AVALIAÇÃO RELEVANTES PARA A APLICAÇÃO ALMEJADA?” Analisando a questão, nota-se que a ontologia passou totalmente pelas duas primeiras avaliações e parcialmente pela terceira. Entretanto, a parte da avaliação que faltou, o enfoque na satisfação do usuário para com a aplicação, deve-se a uma impossibilidade técnica visto que tal requisito só pode ser satisfeito ao final da fase de Aplicação. Este não chega a ser um problema, posto que o próprio autor da OTKM, coloca a avaliação pelo usuário como um “passo em aberto”, sendo que o engenheiro deve decidir a continuidade do projeto baseado nos critérios considerados mais relevantes para a aplicação almejada. Sendo o que se coloca, a decisão é **SIM** – prosseguir.

3.1.5 Aplicação da Fase Aplicação e Evolução

A **evolução** da ontologia ocorreu, como é comum acontecer, quando da implementação da aplicação, visto que alguns de seus elementos tiveram de ser adaptados, visando uma melhor adequação à tecnologia envolvida bem como à lógica do domínio. Podem-se resumir as mudanças ocorridas em:

- agregamento dos sub-conceitos em partições;
- sub-conceitos **Endereco**, **Cidade** e **UF** transformados em atributos do conceito **Local Físico**;
- novos atributos em alguns conceitos;
- propriedades **partOf** substituídas por propriedades relacionais específicas para cada par de classes relacionadas;
- novas propriedades, criadas visando um relacionamento específico entre cada par de conceitos em que há relação;
- novo conceito **Competencia**, particionado em dois novos sub-conceitos, **Criminal** e **Cível**, visando especificar a competencia de cada **Recurso**;
- novos sub-conceitos criados para as partições dos conceitos **Recurso**, **Papel Jurisprudencia**, **Recorrido** e **Recorrente**, descobertos durante o processamento do corpus;
- especificação de sub-propriedades, visando o relacionamento entre conceitos e sub-conceitos.

A evolução percebida, se confrontada com a perguntada ao final desta fase, “HOUVE MUDANÇA?”, leva à resposta **SIM**. Neste caso, a OTKM direciona o ciclo do desenvolvimento para trás, retornando assim à terceira fase, Refinamento. Assim, obteve-se um terceiro refinamento da ontologia semi-formal, aqui representado pela taxonomia apresentada nas Figuras 17 e 18.

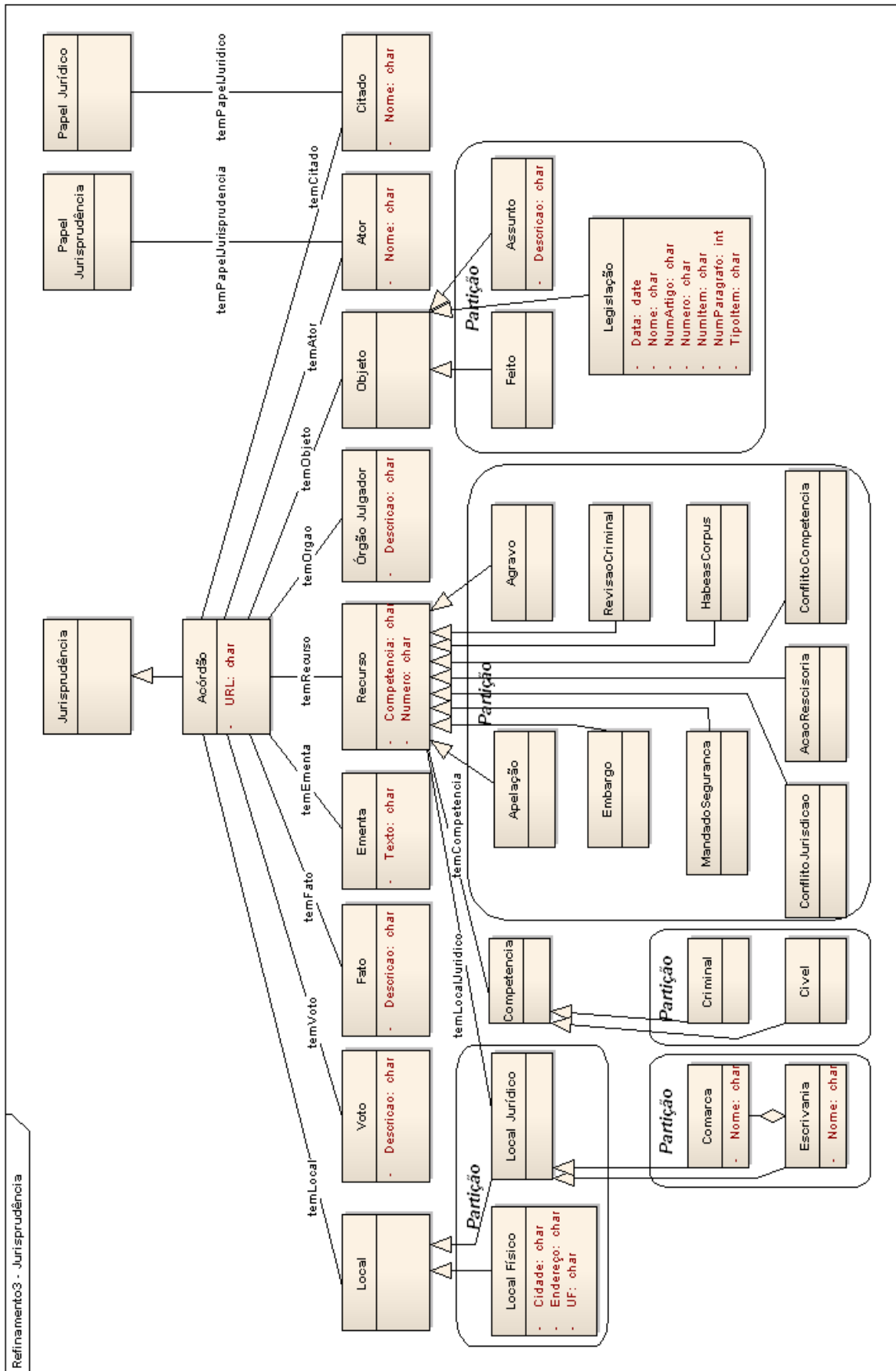


Figura 17: Terceiro refinamento da ontologia semi-formal

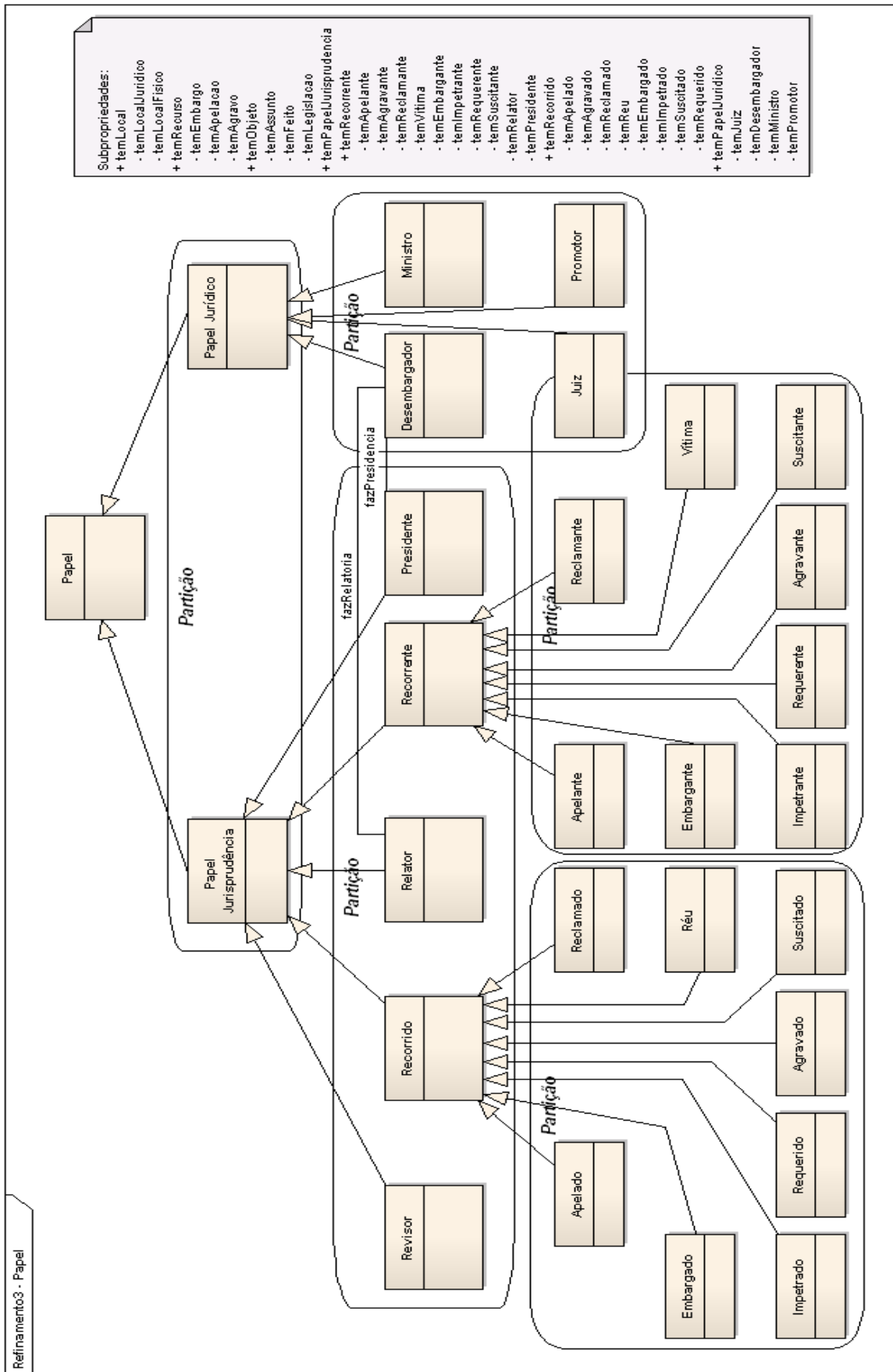


Figura 18: Terceiro refinamento da ontologia semi-formal (continuação)

O novo refinamento da ontologia semi-formal deu subsídios para a criação da versão final da ontologia formal. Note-se que a ontologia resultante não pretende encerrar o domínio completo da jurisprudência. A **ontologia almejada**, desde o início, visa tão somente a **atender aos requisitos da aplicação** neste trabalho postulada.

Assim, é possível agora definir cada conceito através de relações de subsunção e disjunção, bem como de afirmações e axiomas, tudo isso utilizando como linguagem de representação a DL, resultando, por fim, nas expressões a seguir.

Acordao: o conceito é definido pela conjunção de algumas restrições existenciais, as quais indicam que um acórdão **deve** referenciar pelo menos um de cada conceito de Ator, Ementa, Recurso e Voto e várias restrições universais que definem que um acórdão **pode** referenciar os seguintes conceitos: Assunto, Citado, Fato, Legislação, Local, Objeto e Órgão. Por fim, Acordao é subsumido pelo conceito Jurisprudencia, ficando assim a expressão:

$$\begin{aligned} \text{Acordao} &\sqsubseteq \text{Jurisprudencia} \\ \text{Acordao} &\equiv \forall \text{temAssunto.Assunto} \sqcap \exists \text{temAtor.Ator} \sqcap \\ &\quad \forall \text{temCitado.Citado} \sqcap \exists \text{temEmenta.Ementa} \sqcap \\ &\quad \forall \text{temFato.Fato} \sqcap \forall \text{temLegislacao.Legislacao} \sqcap \\ &\quad \forall \text{temLocal.Local} \sqcap \forall \text{temObjeto.Objeto} \sqcap \\ &\quad \forall \text{temOrgao.Orgao} \sqcap \exists \text{temRecurso.Recurso} \sqcap \\ &\quad \exists \text{temVoto.Voto} \end{aligned}$$

Competencia: o conceito é definido como sendo referenciado por pelo menos um recurso, além de possuir um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de Cível ou Criminal, os quais por sua vez são disjuntos, ficando assim a expressão:

$$\begin{aligned} \text{Competencia} &\equiv \exists \text{ehCompetenciaDe.Recurso} \sqcap \\ &\quad (\text{Civel} \sqcup \text{Criminal}) \\ \text{Civel} &\sqsubseteq \text{Competencia} \\ \text{Civel} &\sqsubseteq \neg \text{Criminal} \\ \text{Criminal} &\sqsubseteq \text{Competencia} \\ \text{Criminal} &\sqsubseteq \neg \text{Civel} \end{aligned}$$

Ementa e Voto: são conceitos definidos como sendo referenciados por pelo menos um acórdão, ficando assim as expressões:

$$\text{Ementa} \equiv \exists \text{ehEmentaDe.Acordao}$$

$$\text{Voto} \equiv \exists \text{ehVotoDe.Acordao}$$

Fato e OrgaoJulgador: são conceitos que podem ser referenciados por um acórdão, ficando assim as expressões:

$$\text{Fato} \equiv \forall \text{ehFatoDe.Acordao}$$

$$\text{OrgaoJulgador} \equiv \forall \text{ehOrgaoDe.Acordao}$$

Local: a definição do conceito indica que ele pode ser referenciado por um acórdão, além de possuir um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de LocalFisico ou LocalJuridico, o qual por sua vez também possui um axioma de cobertura para os subconceitos de Comarca e Escrivania, os quais são disjuntos, ficando a expressão da seguinte maneira:

$$\text{Local} \equiv \forall \text{ehLocalDe.Acordao} \sqcap (\text{LocalFisico} \sqcup \text{LocalJuridico})$$

$$\text{LocalFisico} \sqsubseteq \text{Local}$$

$$\text{LocalFisico} \sqsubseteq \neg \text{LocalJuridico}$$

$$\text{LocalJuridico} \sqsubseteq \text{Local}$$

$$\text{LocalJuridico} \sqsubseteq \neg \text{LocalFisico}$$

$$\text{LocalJuridico} \equiv (\text{Comarca} \sqcup \text{Escrivania})$$

$$\text{Comarca} \sqsubseteq \text{LocalJuridico}$$

$$\text{Comarca} \sqsubseteq \neg \text{Escrivania}$$

$$\text{Escrivania} \sqsubseteq \text{LocalJuridico}$$

$$\text{Escrivania} \sqsubseteq \neg \text{Comarca}$$

Objeto: a definição do conceito indica que ele pode ser referenciado por um acórdão, além de possuir um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de Assunto, Feito ou Legislação, os quais, por sua vez, também possuem definições de que eles podem ser referenciados por um acórdão, além de serem disjuntos, ficando assim a expressão:

$$\begin{aligned}
 \text{Objeto} &\equiv \forall e h \text{ObjetoDe.Acordao} \sqcap \\
 &\quad (\text{Assunto} \sqcup \text{Feito} \sqcup \text{Legislacao}) \\
 \text{Assunto} &\sqsubseteq \text{Objeto} \\
 \text{Assunto} &\sqsubseteq \neg \text{Feito} \sqcap \neg \text{Legislacao} \\
 \text{Assunto} &\equiv \forall e h \text{AssuntoDe.Acordao} \\
 \text{Feito} &\sqsubseteq \text{Objeto} \\
 \text{Feito} &\sqsubseteq \neg \text{Legislacao} \sqcap \neg \text{Assunto} \\
 \text{Feito} &\equiv \forall e h \text{FeitoDe.Acordao} \\
 \text{Legislacao} &\sqsubseteq \text{Objeto} \\
 \text{Legislacao} &\sqsubseteq \neg \text{Feito} \sqcap \neg \text{Assunto} \\
 \text{Legislacao} &\equiv \forall e h \text{LegislacaoDe.Acordao}
 \end{aligned}$$

Recurso: o conceito pode ser referenciado em pelo menos um acórdão e deve referenciar pelo menos um de cada conceito Comarca, Competencia e Escrivania. Por fim o conceito é definido por um axioma de cobertura, que limita os seus indivíduos a pertencer exclusivamente aos seus subconceitos – os quais, por sua vez, são disjuntos entre si – ficando assim a expressão:

$$\begin{aligned}
 \text{Recurso} &\equiv \exists \text{ehRecursoDe.Acordao} \sqcap \exists \text{temComarca.Comarca} \sqcap \\
 &\quad \exists \text{temCompetencia.Competencia} \sqcap \exists \text{temEscrivania.Escrivania} \sqcap \\
 &\quad (\text{AcaoRescisoria} \sqcup \text{Agravo} \sqcup \text{Apelacao} \sqcup \\
 &\quad \text{ConflitoJuridicao} \sqcup \text{ConflitoCompetencia} \sqcup \text{Embargo} \sqcup \\
 &\quad \text{MandadoSeguranca} \sqcup \text{RevisaoCriminal}) \\
 \text{AcaoRescisoria} &\sqsubseteq \text{Recurso} \\
 \text{AcaoRescisoria} &\sqsubseteq \neg \text{Agravo} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoJuridicao} \sqcap \neg \text{Embargo} \sqcap \\
 &\quad \neg \text{ConflitoCompetencia} \sqcap \neg \text{MandadoSeguranca} \sqcap \neg \text{RevisaoCriminal} \\
 \text{Agravo} &\sqsubseteq \text{Recurso} \\
 \text{Agravo} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoJuridicao} \sqcap \neg \text{Embargo} \sqcap \\
 &\quad \neg \text{ConflitoCompetencia} \sqcap \neg \text{MandadoSeguranca} \sqcap \neg \text{RevisaoCriminal} \\
 \text{Apelacao} &\sqsubseteq \text{Recurso} \\
 \text{Apelacao} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Agravo} \sqcap \neg \text{ConflitoJuridicao} \sqcap \neg \text{Embargo} \sqcap \\
 &\quad \neg \text{ConflitoCompetencia} \sqcap \neg \text{MandadoSeguranca} \sqcap \neg \text{RevisaoCriminal} \\
 \text{ConflitoCompetencia} &\sqsubseteq \text{Recurso} \\
 \text{ConflitoCompetencia} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Agravo} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoJuridicao} \sqcap \\
 &\quad \neg \text{Embargo} \sqcap \neg \text{MandadoSeguranca} \sqcap \neg \text{RevisaoCriminal} \\
 \text{ConflitoJuridicao} &\sqsubseteq \text{Recurso} \\
 \text{ConflitoJuridicao} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Agravo} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoCompetencia} \sqcap \\
 &\quad \neg \text{Embargo} \sqcap \neg \text{MandadoSeguranca} \sqcap \neg \text{RevisaoCriminal} \\
 \text{Embargo} &\sqsubseteq \text{Recurso} \\
 \text{Embargo} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Agravo} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoJuridicao} \sqcap \\
 &\quad \neg \text{ConflitoCompetencia} \sqcap \neg \text{MandadoSeguranca} \sqcap \neg \text{RevisaoCriminal} \\
 \text{MandadoSeguranca} &\sqsubseteq \text{Recurso} \\
 \text{MandadoSeguranca} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Agravo} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoJuridicao} \sqcap \\
 &\quad \neg \text{ConflitoCompetencia} \sqcap \neg \text{Embargo} \sqcap \neg \text{RevisaoCriminal} \\
 \text{RevisaoCriminal} &\sqsubseteq \text{Recurso} \\
 \text{RevisaoCriminal} &\sqsubseteq \neg \text{AcaoRescisoria} \sqcap \neg \text{Agravo} \sqcap \neg \text{Apelacao} \sqcap \neg \text{ConflitoJuridicao} \sqcap \\
 &\quad \neg \text{ConflitoCompetencia} \sqcap \neg \text{Embargo} \sqcap \neg \text{MandadoSeguranca}
 \end{aligned}$$

Ator: o conceito é definido pela conjunção de duas sentenças: um ator deve ser referenciado em pelo menos um acórdão e deve ter pelo menos um papel de jurisprudência no acórdão, ficando assim a expressão:

$$\text{Ator} \equiv \exists \text{ehAtorDe.Acordao} \sqcap \exists \text{temPapelJurisprudencia.PapelJurisprudencia}$$

Citado: o conceito é definido pela conjunção de duas sentenças: um citado pode ser referenciado em um acórdão e deve ter pelo menos um papel de jurídico no acórdão, ficando assim a expressão:

$$\text{Citado} \equiv \forall \text{ehCitadoPor.Acordao} \sqcap \exists \text{temPapelJuridico.PapelJuridico}$$

Papel: a definição do conceito consiste em um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de PapelJuridico ou PapelJurisprudencia, ficando assim a expressão:

$$\text{Papel} \equiv \text{PapelJuridico} \sqcup \text{PapelJurisprudencia}$$

PapelJuridico: é um subconceito de Papel, disjunto do conceito PapelJurisprudencia e que pode ser referenciado por um Citado. Possui um axioma de cobertura que limita que os seus indivíduos pertençam somente aos seus subconceitos Juiz, Ministro, Promotor ou Desembargador, os quais são disjuntos entre si. Além disso o subconceito Desembargador possui uma definição que indica que ele pode ser referenciado por um Presidente e por um Relator, ficando assim a expressão:

$$\begin{aligned} \text{PapelJuridico} &\sqsubseteq \text{Papel} \\ \text{PapelJuridico} &\sqsubseteq \neg \text{PapelJurisprudencia} \\ \text{PapelJuridico} &\equiv \forall \text{ehPapelJuridicoDe.Citado} \sqcap \\ &\quad (\text{Juiz} \sqcup \text{Ministro} \sqcup \text{Promotor} \sqcup \text{Desembargador}) \\ \text{Juiz} &\sqsubseteq \text{PapelJuridico} \\ \text{Juiz} &\sqsubseteq \neg \text{Desembargador} \sqcap \neg \text{Ministro} \sqcap \neg \text{Promotor} \\ \text{Ministro} &\sqsubseteq \text{PapelJuridico} \\ \text{Ministro} &\sqsubseteq \neg \text{Juiz} \sqcap \neg \text{Desembargador} \sqcap \neg \text{Promotor} \\ \text{Promotor} &\sqsubseteq \text{PapelJuridico} \\ \text{Promotor} &\sqsubseteq \neg \text{Juiz} \sqcap \neg \text{Ministro} \sqcap \neg \text{Desembargador} \\ \text{Desembargador} &\sqsubseteq \text{PapelJuridico} \\ \text{Desembargador} &\sqsubseteq \neg \text{Juiz} \sqcap \neg \text{Ministro} \sqcap \neg \text{Promotor} \\ \text{Desembargador} &\equiv \forall \text{fazPresidencia.Presidente} \sqcap \forall \text{fazRelatoria.Relator} \end{aligned}$$

PapelJurisprudencia: é um subconceito de Papel, disjunto do conceito PapelJuridico, que deve ser referenciado por pelo menos um Ator e possui um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de Presidente, Relator, Revisor, Recorrente ou Recorrido, ficando assim a expressão:

$$\begin{aligned} \text{PapelJurisprudencia} &\sqsubseteq \text{Papel} \\ \text{PapelJurisprudencia} &\sqsubseteq \neg \text{PapelJuridico} \\ \text{PapelJurisprudencia} &\equiv \exists \text{ehPapelJurisprudenciaDe.Ator} \square \\ &\quad (\text{Presidente} \sqcup \text{Relator} \sqcup \text{Revisor} \sqcup \\ &\quad \text{Recorrente} \sqcup \text{Recorrido}) \end{aligned}$$

Presidente e Relator: a definição dos conceitos indica que eles podem ser referenciados por um Desembargador, além de serem subsumidos pelo conceito PapelJurisprudencia. Por fim, ele são disjuntos dos demais subconceitos de PapelJurisprudencia, ficando assim as expressões:

$$\begin{aligned} \text{Presidente} &\sqsubseteq \text{PapelJurisprudencia} \\ \text{Presidente} &\sqsubseteq \neg \text{Recorrente} \square \neg \text{Recorrido} \square \neg \text{Relator} \square \neg \text{Revisor} \\ \text{Presidente} &\equiv \forall \text{ehPresidenciaPor.Desembargador} \\ \text{Relator} &\sqsubseteq \text{PapelJurisprudencia} \\ \text{Relator} &\sqsubseteq \neg \text{Recorrente} \square \neg \text{Recorrido} \square \neg \text{Presidente} \square \neg \text{Revisor} \\ \text{Relator} &\equiv \forall \text{ehRelatadoPor.Desembargador} \end{aligned}$$

Revisor: a definição do conceito indica que ele é subsumido pelo conceito PapelJurisprudencia e é disjunto dos conceitos de Recorrente, Recorrido, Presidente e Relator, ficando assim a expressão:

$$\begin{aligned} \text{Revisor} &\sqsubseteq \text{PapelJurisprudencia} \\ \text{Revisor} &\sqsubseteq \neg \text{Recorrente} \square \neg \text{Recorrido} \square \neg \text{Presidente} \square \neg \text{Relator} \end{aligned}$$

Recorrente: a definição do conceito indica que ele é subsumido pelo conceito PapelJurisprudencia, é disjunto dos conceitos Relator, Recorrido, Presidente e Revisor, e possui um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de Agravante, Apelante, Embargante, Impetrante, Interessado, Reclamante, Suscitante, Vitima ou Requerente – os quais, por sua vez, são disjuntos entre si – ficando assim a expressão:

Recorrente	⊆	PapelJurisprudencia
Recorrente	⊆	¬Relator ⊓ ¬Recorrido ⊓ ¬Presidente ⊓ ¬Revisor
Recorrente	≡	(Agravante ⊓ Apelante ⊓ Embargante ⊓ Impetrante ⊓ Interessado ⊓ Reclamante ⊓ Suscitante ⊓ Vitima ⊓ Requerente)
Agravante	⊆	Recorrente
Agravante	⊆	¬Apelante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Vitima ⊓ ¬Requerente
Apelante	⊆	Recorrente
Apelante	⊆	¬Agravante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Vitima ⊓ ¬Requerente
Embargante	⊆	Recorrente
Embargante	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Vitima ⊓ ¬Requerente
Impetrante	⊆	Recorrente
Impetrante	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Embargante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Vitima ⊓ ¬Requerente
Interessado	⊆	Recorrente
Interessado	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Vitima ⊓ ¬Requerente
Reclamante	⊆	Recorrente
Reclamante	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Suscitante ⊓ ¬Vitima ⊓ ¬Requerente
Suscitante	⊆	Recorrente
Suscitante	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Vitima ⊓ ¬Requerente
Vitima	⊆	Recorrente
Vitima	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Requerente
Requerente	⊆	Recorrente
Requerente	⊆	¬Agravante ⊓ ¬Apelante ⊓ ¬Embargante ⊓ ¬Impetrante ⊓ ¬Interessado ⊓ ¬Reclamante ⊓ ¬Suscitante ⊓ ¬Vitima

Recorrido: a definição do conceito indica que ele é subsumido pelo conceito PapelJurisprudencia, é disjunto dos conceitos Relator, Recorrente, Presidente e Revisor, e possui um axioma de cobertura que limita que os seus indivíduos pertençam somente aos subconceitos de Agravado, Apelado, Embargado, Impetrado, Reclamado, Suscitado, Reu ou Requerido – os quais, por sua vez, são disjuntos entre si – ficando assim a expressão:

$$\begin{aligned}
 \text{Recorrido} &\sqsubseteq \text{PapelJurisprudencia} \\
 \text{Recorrido} &\sqsubseteq \neg \text{Relator} \sqcap \neg \text{Recorrente} \sqcap \neg \text{Presidente} \sqcap \neg \text{Revisor} \\
 \text{Recorrido} &\equiv (\text{Agravado} \sqcup \text{Apelado} \sqcup \text{Embargado} \sqcup \text{Impetrado} \sqcup \\
 &\quad \text{Reclamado} \sqcup \text{Suscitado} \sqcup \text{Reu} \sqcup \text{Requerido}) \\
 \text{Agravado} &\sqsubseteq \text{Recorrido} \\
 \text{Agravado} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Reu} \sqcap \neg \text{Requerido} \\
 \text{Apelado} &\sqsubseteq \text{Recorrido} \\
 \text{Apelo} &\sqsubseteq \neg \text{Agravado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Reu} \sqcap \neg \text{Requerido} \\
 \text{Embargado} &\sqsubseteq \text{Recorrido} \\
 \text{Embargado} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Agravado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Reu} \sqcap \neg \text{Requerido} \\
 \text{Impetrado} &\sqsubseteq \text{Recorrido} \\
 \text{Impetrado} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Agravado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Reu} \sqcap \neg \text{Requerido} \\
 \text{Reclamado} &\sqsubseteq \text{Recorrido} \\
 \text{Reclamado} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Agravado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Reu} \sqcap \neg \text{Requerido} \\
 \text{Suscitado} &\sqsubseteq \text{Recorrido} \\
 \text{Suscitado} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Agravado} \sqcap \neg \text{Reu} \sqcap \neg \text{Requerido} \\
 \text{Reu} &\sqsubseteq \text{Recorrido} \\
 \text{Reu} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Agravado} \sqcap \neg \text{Requerido} \\
 \text{Requerido} &\sqsubseteq \text{Recorrido} \\
 \text{Requerido} &\sqsubseteq \neg \text{Apelado} \sqcap \neg \text{Embargado} \sqcap \neg \text{Impetrado} \sqcap \\
 &\quad \neg \text{Reclamado} \sqcap \neg \text{Suscitado} \sqcap \neg \text{Reu} \sqcap \neg \text{Agravado}
 \end{aligned}$$

A corrente fase ainda almeja a construção da **aplicação** de gestão da ontologia, o que será abordado, dada a sua complexidade e extensão, em uma seção específica, 3.2.

Por fim, chegou-se ao resultado esperado para a corrente fase: a ontologia evoluída. Com isso analisa-se novamente a pergunta ao final, “HOUVE MUDANÇA?”, cuja resposta, neste ponto, é **NÃO**, podendo-se assim seguir em frente para a construção da aplicação que se baseará na ontologia resultante. Note-se que as respostas dadas às perguntas, analisadas ao final de cada fase, foram inferidas baseado tão somente no senso do autor, visto que a OTKM não define, e nem exige, métricas para se obter as respostas de maneira lógica.

3.2 CONSTRUÇÃO DA APLICAÇÃO DE GESTÃO DA ONTOLOGIA

O Metaprocesso culminou com a criação e avaliação da ontologia JurisTJPR. Assim, é possível passar para a última fase deste processo, a qual visa a aplicar e manter a ontologia resultante. O principal requisito do projeto é a extração de instâncias de conceitos, sub-conceitos e propriedades de um corpus de acórdãos, visando à sua utilização na população da ontologia JurisTJPR. Assim, neste trabalho optou-se pela construção de um aplicativo, dada a necessidade de extração de informações, o que não seria possível através de sistemas prontos de mineração de dados, como, por exemplo, o Weka (FRANK et al., 2009).

O primeiro requisito, técnico, para a construção do aplicativo foi a sua codificação utilizando a linguagem Java, por ter a sua licença de utilização regida por regras de software livre; por ser a linguagem utilizada na codificação de todas as bibliotecas de códigos abordadas na revisão bibliográfica; por ter se tornado, no momento, uma linguagem padrão para qualquer aplicação acadêmica, facilitando a reutilização do código em trabalhos futuros.

As técnicas e ferramentas utilizadas para o desenvolvimento, dentre aquelas apresentadas na revisão bibliográfica, capítulo 2 foram:

Extração de Instâncias de Conceitos: para a extração foi utilizada a **RegEx**, biblioteca para análise de expressões regulares, disponível no pacote *java.util* do Java2, a partir da versão 1.4;

Stemming: o processamento de *stemming* efetuado nas sentenças extraídas se deu através da biblioteca PortugueseStemmer;

Avaliação de relevância de termos: para este requisito optou-se por busca de similaridade com os descritores existentes no Tesouro Jurídico da Justiça Federal;

Representação e armazenamento da ontologia: foi utilizada a linguagem OWL-DL, uma sub-linguagem da OWL que implementa a expressividade da Lógica de Descrições;

Gestão da ontologia: a gestão (carga em memória, navegação, população e gravação em arquivo) da ontologia foi implementada baseada na *framework* Jena.

Assim, as metodologias, técnicas e ferramentas até aqui abordadas, por fim promovem o embasamento tecnológico necessário para se construir a aplicação almejada neste trabalho, a qual foi planejada, para quebrar a sua complexidade, em cinco módulos:

Extração de Corpus é um módulo externo à aplicação, o qual foi executado previamente, visando à composição do corpus, cujos documentos necessitavam ser extraídos de sua base de dados de origem;

Visualização é o módulo de entrada e saída do aplicativo, cuja função é proceder à carga da ontologia do arquivo para a memória, representá-la graficamente na tela do computador e, por fim, gravá-la novamente no arquivo, visando a tornar permanentes as alterações efetuadas na ontologia em memória. Para a implementação deste módulo foi reaproveitado o sistema TreeVis, de autoria de Anderson Seiti Yamamoto, construído na plataforma Java, sob a orientação do docente pesquisador deste programa de pós-graduação, Dr. Cesar Augusto Tacla, visando justamente à carga, à visualização (em forma de árvore) e à gravação de ontologias;

Extração é o módulo onde se efetua uma varredura do corpus, visando à extração de instâncias para os conceitos de jurisprudência mapeados na ontologia. O resultado deste módulo é gravado em memória em uma lista de conceito-instância;

População é o módulo onde a ontologia, residente na memória principal, é povoada com as instâncias extraídas através do módulo anterior, com o qual trabalha iterativamente;

Pesquisa é o módulo onde a ontologia povoada, presente na memória principal, pode ser consultada através de consultas informadas em alto nível pelo usuário e convertidas na linguagem de consulta SPARQL.

Assim, o sistema TreeVis, adicionado dos módulos de extração de instâncias de conceitos e de população da ontologia JurisTJPR, recebeu um novo nome: jurTreeVis. Deve-se esclarecer que este aplicativo consiste em um protótipo, o qual visa, tão somente, à aplicação dos métodos aqui propostos bem como provar a viabilidade deles, sendo que não há aqui o compromisso em contemplar todas as possibilidades e situações possíveis de se encontrar no domínio da jurisprudência, o que seria uma restrição para um sistema a ser colocado em produção. Os componentes criados ou utilizados para compor a solução podem ser visualizados conforme a Figura 19, sendo detalhados nas subseções a seguir, organizados por módulos.

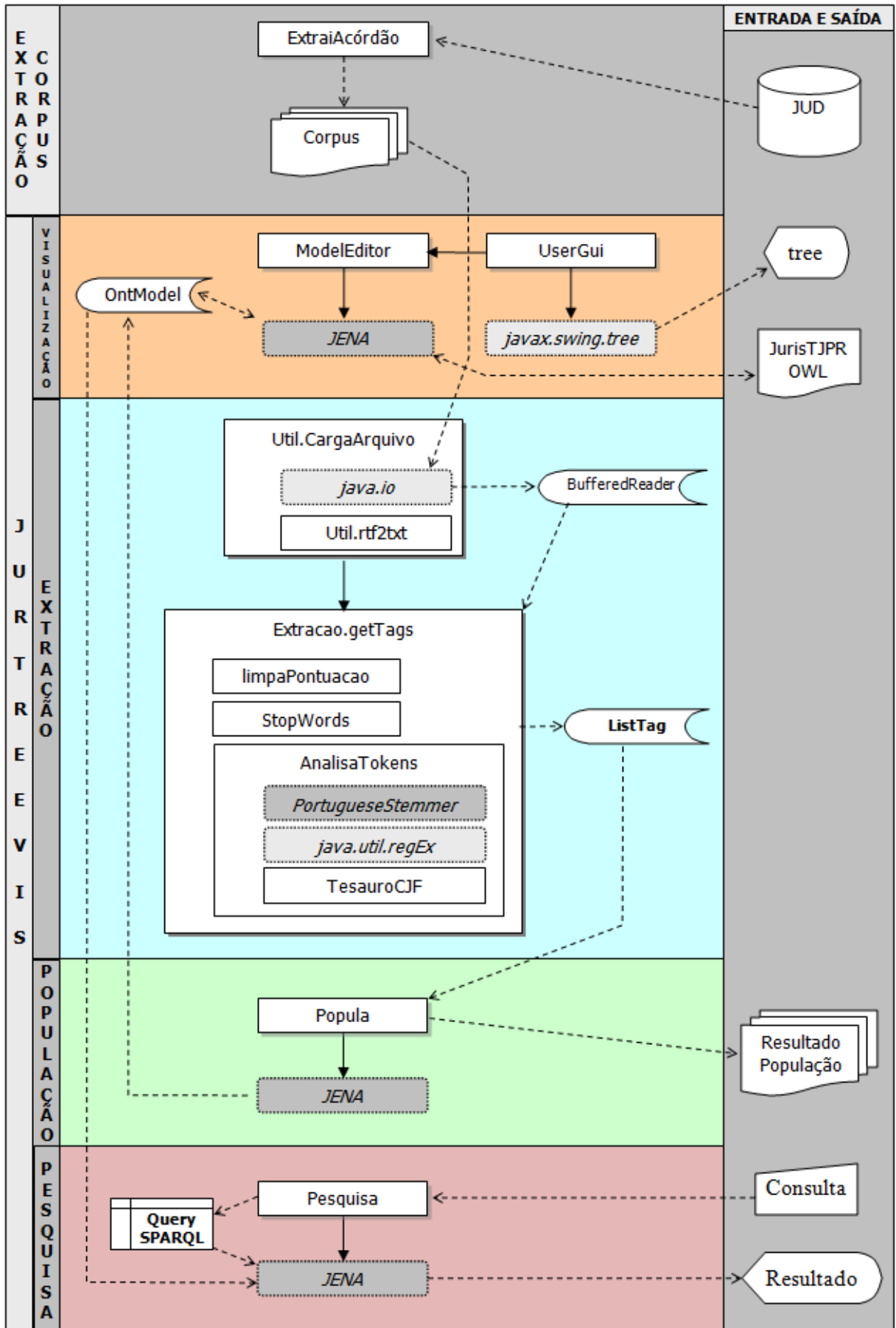


Figura 19: Componentes da aplicação

3.2.1 Módulo de Extração de Corpus

Trata-se de um componente externo à aplicação, o qual visa tão somente a obter os documentos de acórdãos que compõem o *Corpus* que servirá de entrada, sendo assim executado apenas uma vez. Os documentos encontram-se na base de dados de testes *JUD*, residente em um Sistema Gerenciador de Base de Dados (SGBD) Sybase, a qual contém os dados resultantes do sistema de controle processual Judwin, incluindo os acórdãos proferidos pelos desembargadores do TJPR. Os documentos, elaborados em formato Rich Text Format (RTF), são gravados em formato binário na tabela *AcordaoTexto*, cuja estrutura se apresenta conforme a Figura 20.

```
CREATE TABLE dbo.AcordaoTexto (
    OrgJul      smallint NOT NULL,
    Acordao     int       NOT NULL,
    Processo    int       NOT NULL,
    Texto       image     NULL
)
```

Figura 20: Estrutura da tabela de documentos de acórdãos

Assim, foi construído um pequeno programa, *ExtraiAcordao*, em Delphi (plataforma escolhida visando a reaproveitar código já existente) o qual carrega em memória cada documento existente na citada tabela e o grava no sistema de arquivos do Windows.

3.2.2 Módulo de Visualização

No módulo de visualização a classe *ModelEditor* invoca a API do *Jena* para efetuar a carga dos conceitos e propriedades, e respectivas instâncias, da ontologia que for fornecida como entrada ao sistema. Arquivos no formato OWL ou RDF, de qualquer ontologia, podem ser oferecidos neste momento, visando à sua visualização. Assim, a carga da ontologia em memória resulta na criação do objeto *Model* da classe *OntModel*, originária da biblioteca *Jena*.

Na sequência, a classe *UserGui* é invocada para a composição e saída em tela de uma árvore hierárquica (*tree*) que representa a ontologia carregada, o que é feito utilizando as classes básicas da biblioteca do Java *javax.swing.tree*, conforme pode-se ver na Figura 21,

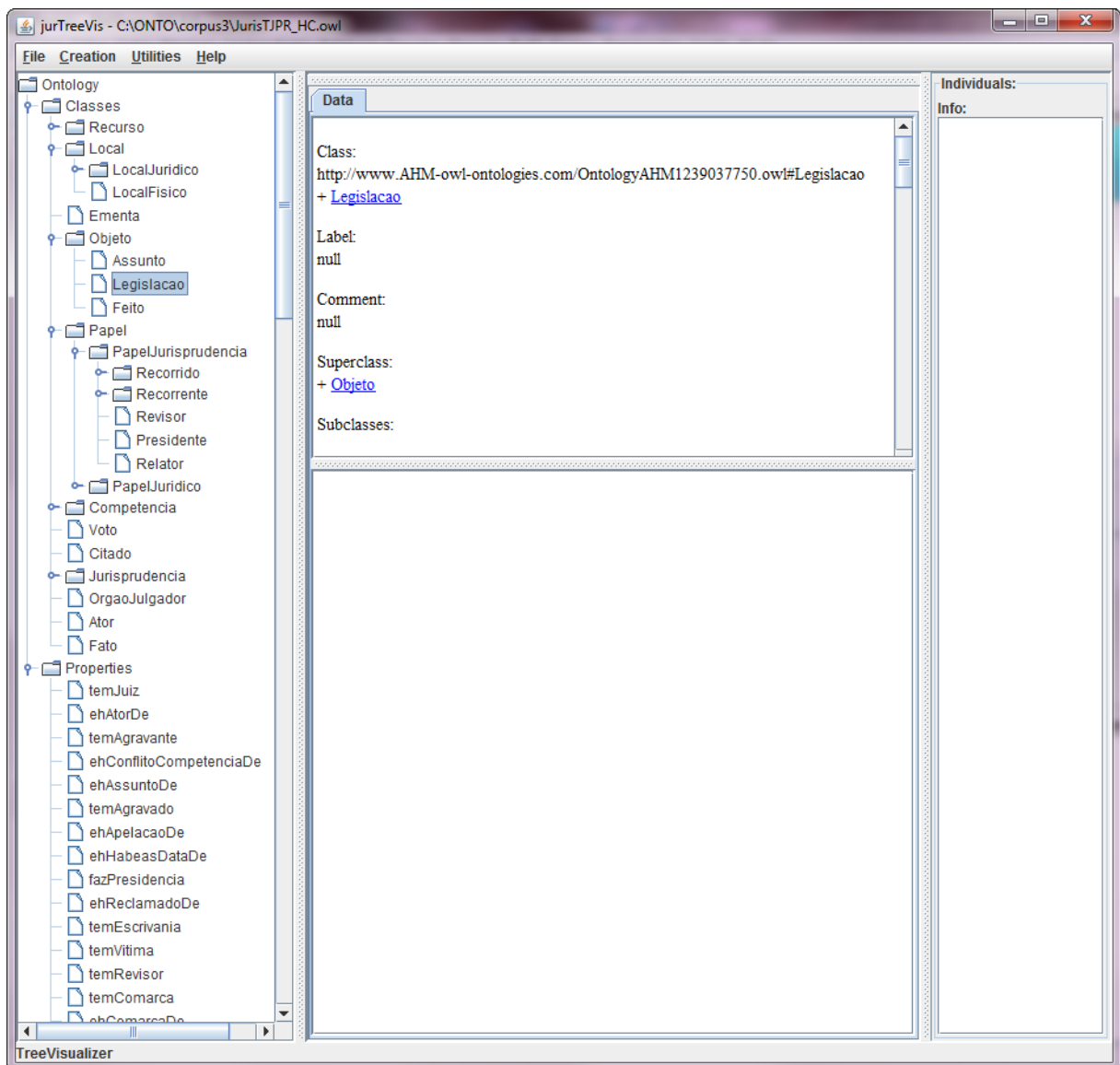


Figura 21: Janela do módulo de visualização do jurTreeVis

onde:

- o quadro da esquerda demonstra, na forma de árvore hierárquica, a taxonomia das *Classes*, ou conceitos, da ontologia carregada, bem como as suas propriedades;
- o quadro superior central, intitulado *Data*, apresenta detalhes da classe selecionada: URI do conceito (*Class*), rótulo (*Label*), comentários (*Comment*), conceitos genéricos (*Superclass*) e conceitos específicos (*Subclasses*);
- os demais quadros serão abordados mais adiante;

3.2.3 Módulo de Extração

Este módulo, resumidamente, abre os arquivos do corpus e submete-os a métodos de extração, gerando modelos XML que contém os indivíduos extraídos de cada documento. Deve-se aqui esclarecer que, apesar de o módulo de visualização possibilitar a carga em memória e a representação gráfica de qualquer modelo contido em arquivo RDF ou OWL, o módulo de extração, bem como os seguintes descritos, só terá efeito quando a ontologia JurisTJPR estiver em memória.

O jurTreeVis utiliza um método iterativo de extração e população, onde, a cada iteração, um arquivo é carregado, processado e os seus indivíduos populados na ontologia. Tais funcionalidades são ativadas através da janela intitulada *População*, conforme a Figura 22,

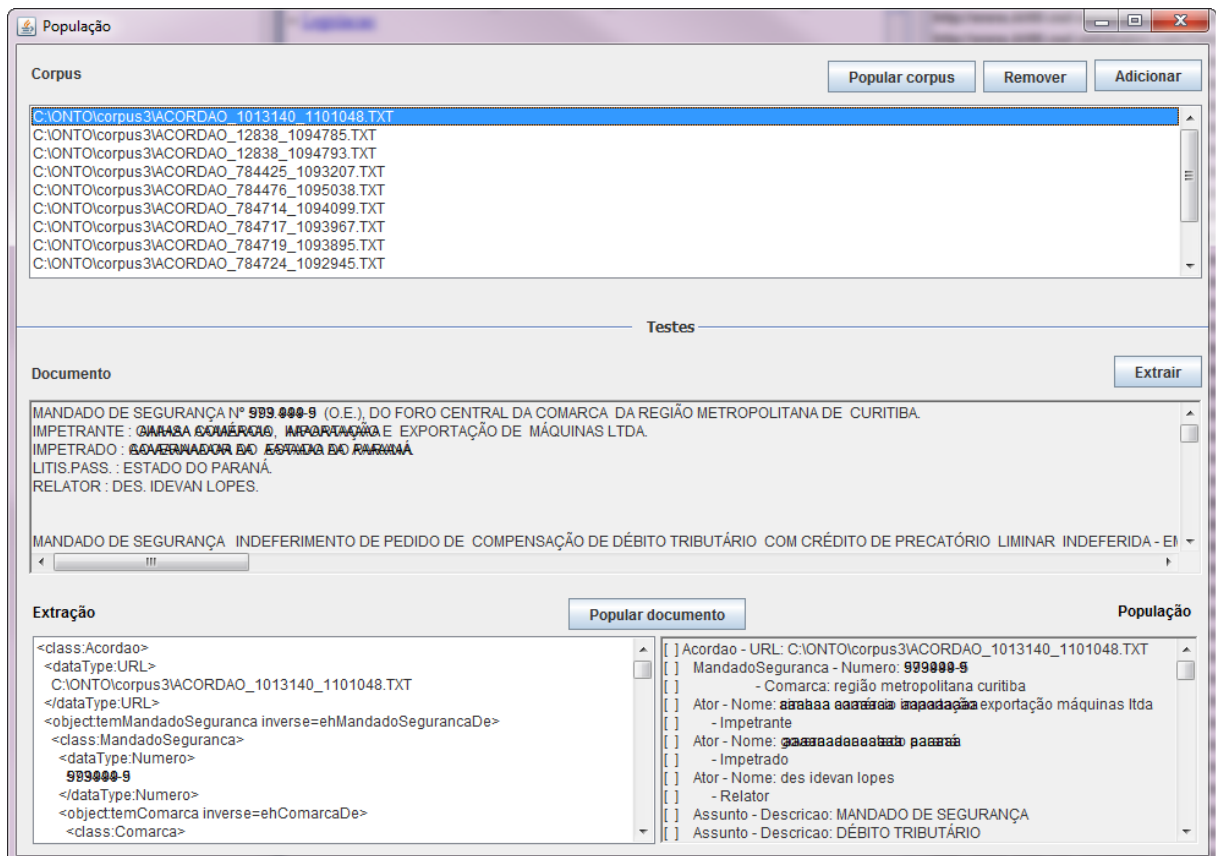


Figura 22: Janela de extração/população da ontologia

onde:

Corpus é uma caixa contendo a lista dos documentos selecionados do corpus que serão processados;

Adicionar é um botão através do qual se podem selecionar os documentos que se deseja processar. Os arquivos selecionados tem seus nome e caminho no sistema de arquivos adicionados à caixa *Corpus*;

Remove é um botão através do qual se pode excluir a seleção de um arquivo da caixa *Corpus*;

Popular corpus é o botão que efetua o método de extração e população tendo como entrada os arquivos informados na caixa *Corpus*. Ao final do processamento, os indivíduos extraídos do corpus estarão todos populados na ontologia, a qual foi carregada para a memória através do módulo de visualização;

Testes é a área da janela que dispõe de componentes que possibilitam testes de extração e população com um documento (aquele que estiver selecionado na caixa *Corpus*);

Documento é uma caixa onde se apresenta o conteúdo do documento que estiver selecionado na caixa *Corpus*;

Extrair é um botão através do qual se processam os métodos de extração sobre o texto contido na caixa *Documento*;

Extração é uma caixa onde se apresenta o modelo XML resultante dos métodos de extração;

Popular documento é um botão através do qual se processa o método de população da ontologia, residente em memória, tendo como entrada o modelo XML contido na caixa *Extração*;

População é uma caixa onde se apresenta um sumário da população efetuada na ontologia, através de um texto plano contendo os nomes dos conceitos ou propriedades populados, seguidos dos respectivos indivíduos;

Os métodos de extração se iniciam com o método *Util.CargaArquivo* o qual efetua a carga do próximo arquivo para a memória, através das classes básicas da biblioteca *java.io*.

Na sequência o método *Util.rtf2txt* transforma o formato do conteúdo lido do arquivo, de RTF, para texto plano, através da biblioteca *javax.swing.text.DefaultStyledDocument*, visando a remover os caracteres de controle daquele formato.

Seguindo o fluxo, chega-se ao ponto principal deste módulo, a execução do método *Extracao.getTags*, o qual efetua a extração de termos e sentenças contidas no documento carregado – através de localizações, sinais ou expressões regulares – que possam vir a ser indivíduos de conceitos e propriedades da jurisprudência, retornando-os em tags sob um formato definido pelo autor, semelhante, em essência, ao RDF, conforme o Quadro 7.

Tipo de componente	Tag
Conceito	<class:nomeConceito> </class:nomeConceito>
Propriedade tipo dado	<dataType:nomePropriedade> </dataType:nomePropriedade>
Propriedade tipo objeto	<object:nomePropriedade inverse=nomePropriedadeInversa> </object:nomePropriedade>

Quadro 7: Definição de tags para representação de extração de conceitos

O método *getTags* é iterativo, ou seja, para cada linha existente no documento são efetuados todos os passos necessários para o processamento da extração de instâncias, resultando em todos os conceitos naquela linha encontrados. Entenda-se por linha: uma cadeia de caracteres quaisquer terminados por um caractere indicador de fim-de-linha.

Assim, a iteração se inicia pela execução do método **limpaPontuacao**, o qual remove da linha todas as pontuações e símbolos que possam existir no texto. Ainda neste método, a varredura é aproveitada para adiantar uma tarefa simples de *stemming*, a eliminação de pronomes pessoais de terceira pessoa sufixados a verbos, a saber: **o(s)**, **a(s)**, **lo(s)**, **la(s)**, **se** e **lhe(s)**.

Na sequência é efetuada a remoção das *stop words*, através da classe Java **StopWords**, a qual consiste em uma adaptação da biblioteca *StopwordsEnglish* (FRANK, 2010) em cujo código foi inserida a tabela de palavras da língua portuguesa de Savoy (2010) à qual foram acrescentadas variações sem acento de stop-words acentuadas. Na biblioteca *AnalisaTokens*, invocada em seguida, é onde, por fim, acontece a extração das sentenças a serem conceitualizadas. Há um método para cada conceito almejado, mapeando as suas particularidades através de, pelo menos, uma de suas características de localização, sinalização e ocorrência, conforme demonstrado no Quadro 8.

Método	Localização	Ocorrência	Sinalização	Expr. regular	Exemplo de tags resultantes
RECURSO	<i>Linha 1</i>		0-9 . - / recurso apelacao embargo mandado acao revisao conflito habeas agravo cível criminal	[0-9[\u002E] [\u002D] [\u002F]]+	<object:temRecurso inverse=ehRecursoDe> <class:Recurso> <object:temCompetencia> <class:Cível> </class:Cível> </object:temCompetencia> <dataType:Numero>

(continua)

Quadro 8: Características abordadas nos métodos de extração

Método	Localização	Ocorrência	Sinalização	Expr. regular	Exemplo de tags resultantes
					379178-6 </dataType:Numero> </class:Recurso> </object:temRecurso>
COMARCA	<i>Linha 1</i> + comarca		a-z !vara !juizado !escrivania !oficio	[a-z[A-Z]]+	<object:temComarca inverse=ehComarcaDe> <class:Comarca> <dataType:Nome> apucarana </dataType:Nome> </class:Comarca> </object:temComarca>
ESCRIVANIA	<i>Linha 1</i> + vara <i>ou</i> juizado <i>ou</i> oficio <i>ou</i> escrivania		0-9 ^{o a} . primeir segund terceir quart quint sext setim oitav non decim vigessim trigesim quadragessim quinquagesim sexagesim heptagesim octagesim nonagesim	[0-9[\u00B0 \u00BA \u002E] [\u00AA]]+	<object:temEscrivania inverse=ehEscrivaniaDe> <class:Escrivania> <dataType:Nome> primeira Vara cível </dataType:Nome> </class:Escrivania> </object:temEscrivania>
ATOR			apelante reu recorrente autor embargante impetrante agravad recorrid revisor apelad embargad requerid suscitad		<object:temAtor inverse=ehAtorDe> <class:Ator> <dataType:Nome> banco xxxxxxxxxxx xxxxxxxxxxx s/a </dataType:Nome> <object:temApelante> <class:Apelante> </class:Apelante> </object:temApelante>

(continua)

Quadro 8: Características abordadas nos métodos de extração

Método	Localização	Ocorrência	Sinalização	Expr. regular	Exemplo de tags resultantes
			vítima impetrad suscitante agravante reclamante reclamad interessad requerente		<pre> </class:Ator> </object:temAtor> </pre>
RELATOR			relator		<pre> <object:temAtor inverse=ehAtorDe> <class:Ator> <dataType:Nome> des.^a vilma régia ramos rezende </dataType:Nome> <object:temRelator> <class:Relator> </class:Relator> </object:temRelator> </class:Ator> </object:temAtor> </pre>
VOTO	acordam		conced desprov dar neg recurs indefer total parc part mant provido mantid proced embargo sentenc agravo improced defer acolhido provimento rejeitado		<pre> <object:temVoto inverse=ehVotoDe> <class:Voto> <dataType:Descricao> RECURSO PARCIALMENTE PROVIDO. </dataType:Descricao> </class:Voto> </object:temVoto> </pre>
ORGAO _JULGADOR	acordam		camara decima vigesima trigesima		<pre> <object:temOrgao inverse=ehOrgaoDe> <class:OrgaoJulgador> <dataType:Descricao> </pre>

(continua)

Quadro 8: Características abordadas nos métodos de extração

Método	Localização	Ocorrência	Sinalização	Expr. regular	Exemplo de tags resultantes
			quadragésima quingentesima sexagesima heptagesima octagesima nonagesima		quinta camara civil </dataType:Descricao> </class:OrgaoJulgador> </object:temOrgao>
EMENTA	<i>Próxima linha após a extração de Relator</i>	TesouroCJF .estahContido()	- .	(\u002E\s)(\s-) (\s-) (-\s) (-\s)	<object:temAssunto inverse=ehAssuntoDe> <class:Assunto> <dataType:Descricao> EMBARGOS À EXECUÇÃO </dataType:Descricao> </class:Assunto> </object:temAssunto>
		!TesouroCJF .estahContido()	- .	(\u002E\s)(\s-) (\s-) (-\s) (-\s)	<object:temFato inverse=ehFatoDe> <class:Fato> <dataType:Descricao> CONSTITUCIONAL E TRIBUTÁRIO </dataType:Descricao> </class:Fato> </object:temFato>
VOTOementa	<i>Última sentença da Ementa</i>		conced agravo mant dar provimento neg indefer total acao parc defer mandado provido revisao proced recurs part sentenc improced embargo apelaca mantid acolhido desprov rejeitado		<object:temVoto inverse=ehVotoDe> <class:Voto> <dataType:Descricao> RECURSO PARCIALMENTE PROVIDO. </dataType:Descricao> </class:Voto> </object:temVoto>

(continua)

Quadro 8: Características abordadas nos métodos de extração

Método	Localização	Ocorrência	Sinalização	Expr. regular	Exemplo de tags resultantes
LEGISLACAO	<i>Antes de um dos termos:</i> lei leis lc decreto dl decreto-lei constituicao ctn cpc cp cpp codigo				<object:temLegislacao inverse=ehLegislacaoDe> <class:Legislacao>
			inciso inc anexo caput item itens		<dataType:TipoItem> inciso </dataType:TipoItem>
			IVX	[a-z&&[ivx]]+	<dataType:NumItem> xxxii </dataType:NumItem>
			art art ^o artigo		<dataType:NumArtigo> 5 </dataType:NumArtigo>
			0-9 a-d ° - §	[0-9[a-d][\u00B0 [\u00BA] [\u002D][§]]+	<dataType:NumParagrafo> 2 </dataType:NumParagrafo>
	<i>Depois de um dos termos:</i> lei leis lc decreto dl decreto-lei constituicao ctn cpc cp cpp codigo		tributario nacional fiscais processo execuc penal fiscal consumidor federal estadual organica defesa municipal organica comercial complementar civil public ministerio		<dataType:Nome> código processo civil </dataType:Nome>
			0-9 n . ° /	[0-9[n][\u00B0 [\u00BA] [\u002E] [\u002F]]+	<dataType:Numero> 6.830/80 </dataType:Numero>
			09/09/0099 09.09.0099	\d{1,2}/\d{1,2} /\d{2,4} \d{1,2}.\d{1,2} .\d{2,4}	<dataType:Data> 15/12/1968 </dataType:Data>

(continua)

Quadro 8: Características abordadas nos métodos de extração

Método	Localização	Ocorrência	Sinalização	Expr. regular	Exemplo de tags resultantes
			0-9 0-9 janeiro marco maio abril junho agosto outubro setembro julho novembro fevereiro dezembro	\d{1,2}	<dataType:Data> 15 dezembro 1968 </dataType:Data> </class:Legislacao> </object:temLegislacao>
			0-9 0-9 0-9 0-9	\d{2,4}	

Quadro 8: Características abordadas nos métodos de extração

Onde:

Método indica o nome de cada método (sub-rotina), disponível na biblioteca *AnalisaTokens*, para extração instâncias de conceitos, propriedades e indivíduos;

Localização indica o método de localização do conceito no acórdão, o que é possível graças a uma semi-estrutura que os acórdãos apresentam, fruto de consenso antigo entre os relatores, conforme pode-se ver no documento apresentado no anexo A. As localizações mapeadas foram as seguintes:

- *Linha 1*: indica que o conceito *Recurso*, quando houver no documento, será sempre encontrado na primeira linha do texto;
- *Linha 1 + comarca*: indica que o conceito *Comarca* pode ser encontrado na primeira linha do texto após a ocorrência do termo **comarca**;
- *Linha 1 + vara ou juizado ou officio ou escrivania*: indica que o conceito *Escrivania* pode ser encontrado na primeira linha do texto a partir de um dos termos citados;
- *acordam*: indica que os conceitos *Voto* e *OrgaoJulgador* podem ser encontrados na linha onde ocorra o termo **acordam**;
- *Próxima linha após a extração de Relator*: indica que, uma vez que o conceito *Relator* tenha sido encontrado, a próxima linha com conteúdo será considerada como *Ementa*;

- *Última sentença da Ementa*: indica que na última sentença da *Ementa* localizada poderá haver o conceito *Voto*;
- *Antes e depois de um dos termos lei leis lc código decreto decreto-lei dl constituição ctn cpc cp cpp*: indica que sempre que aparecer no texto um dos termos citados, o seu entorno pode compor o conceito *Legislação*;

Ocorrência indica o método de aferição da ocorrência de termos relevantes para o domínio dos acórdãos, o que neste projeto é aplicável somente ao local da *Ementa*, através de consulta ao Tesouro do CJF, enquadrando cada sentença em dois conceitos possíveis:

- *TesouroCJF.estaContido()* indica que a sentença, ou parte dela, foi encontrada no tesouro, o que nos leva à consideração de que trata-se de assunto recorrente no jargão jurídico, sendo então relevante à todo o conjunto da jurisprudência, passando assim a consistir em um conceito *Assunto*;
- *!TesouroCJF.estaContido()* indica que nenhuma parte da sentença foi encontrada no tesouro, levando à consideração que trata-se de assunto específico do documento e de pouca relevância para o conjunto da jurisprudência, consistindo assim em um conceito *Fato*;

Sinalização são os *bonus* que podem sinalizar a existência de um indivíduo de um conceito esperado ou *stigmas* que podem servir como delimitadores, sendo que estes estão representados pelo sinal “!” (veja o Quadro 8 que se inicia na página 111). Por exemplo, no método *extraiCOMARCA*, uma instância do conceito *Comarca* pode existir na linha 1 do texto após o termo *bonus comarca*, desde de não contenha os termos *stigmas vara, juizado, escritania* ou *ofício*. Quando o sinal consiste em algarismos, letras ou símbolos a sua busca se dá através de uma expressão regular que representa a sua sintaxe, conforme a coluna seguinte do quadro, como por exemplo, no método *RECURSO*, em que os algarismos de 0 a 9, bem como os caracteres ‘.’, ‘-’ e ‘/’, são considerados *bonus* para um indivíduo da propriedade *Numero* do conceito *Recurso*.

Expr. regular apresenta as expressões regulares que foram utilizadas na busca dos sinais procurados pelo corrente método, as quais tem os seguintes significados:

`[0-9[\u002E][\u002D][\u002F]]+` é utilizada no método *RECURSO* visando a expressar um número que pode possuir, além de algarismos, o ponto separador de milhar e os separadores de dígitos complementares - ou /. Este é o caso da propriedade *Numero* do conceito *Recurso*, o qual pode ser composto por um número sequencial seguido de um número de ano, como exemplo: **1.025/69**;

`[a-zA-Z]+` é utilizada no método COMARCA para expressar uma sequência composta exclusivamente por letras, que é o caso da propriedade *Nome* do conceito *Comarca*, por exemplo: **apucarana**;

`[0-9][\u00B0\u00BA\u002E][\u00AA]+` é utilizada no método ESCRIVANIA onde expressa um número ordinal composto por algarismos e símbolos específicos dos ordinais, caso típico da propriedade *Nome* do conceito *Escrivania*, quando parte deste é composto por um ordinal, por exemplo: **5^a. vara criminal**;

`(\u002E\s)|(\s-)|(\s-)(-\s)|(-\s)` é utilizada no método EMENTA para representar um texto onde há sentenças separadas por pontos ou traços. Nesta expressão regular, cada expressão entre parêntesis representa uma possível forma de se encontrar os separadores de sentença, a saber:

- ponto seguido de espaço (`\u002E\s`);
- traço precedido de espaço (`\s-`);
- hífen precedido de espaço (`\s-`) – apesar da aparente redundância com a alínea anterior, o traço tem uma codificação distinta do hífen na tabela ASCII, sendo que, dependendo da edição do texto, os dois caracteres podem se alternar entre os textos;
- traço seguido de espaço (`-\s`);
- hífen seguido de espaço (`-\s`).

Como exemplo de ementa tem-se “EMBARGOS À EXECUÇÃO. CÉDULA DE CRÉDITO COMERCIAL - TÍTULO REVESTIDO DOS REQUISITOS LEGAIS-REDUZIU-SE O PERCENTUAL DA MULTA PARA 2% -PROVIMENTO PARCIAL COM NOVA DEFINIÇÃO SUCUMBENCIAL.”, onde as sentenças distinguidas pela expressão regular são:

- EMBARGOS À EXECUÇÃO
- CÉDULA DE CRÉDITO COMERCIAL
- TÍTULO REVESTIDO DOS REQUISITOS LEGAIS
- REDUZIU-SE O PERCENTUAL DA MULTA PARA 2%
- PROVIMENTO PARCIAL COM NOVA DEFINIÇÃO SUCUMBENCIAL.

`[a-z&&[ivx]]+` é utilizada no método LEGISLACAO, representando uma sequência composta exclusivamente pelas letras i, v ou x, que compõem um número romano, como no caso da propriedade *NumItem* do conceito *Legislacao*. Como exemplo: **xxxii**;

`[0-9[a-d][\u00B0][\u00BA][\u002D][§]]+` é também utilizada no método LEGISLACAO visando a expressar um número ordinal de parágrafo que pode ser seguido de alínea

designada por uma letra, por exemplo: **10º**, **§3º**, **54-B**. Este é o caso da propriedade *NumParagrafo* do conceito *Legislacao*;

`[0-9[n][\u00B0][\u00BA][\u002E][\u002F]]+` é bastante semelhante à primeira expressão acima descrita, com o diferencial que nesta se admite também os símbolos ‘n’ e ‘º’ de ordinais. Esta expressão é utilizada no método LEGISLACAO sendo o caso da propriedade *Numero* do conceito *Legislacao*, como exemplo: **6.830/80**.

`\d{1,2}/\d{1,2}/\d{2,4}` é utilizada no método LEGISLACAO para representar uma data com barras separando o dia do mês e do ano, caso da propriedade *Data* do conceito *Legislacao*, por exemplo: **15/12/1968**;

`\d{1,2}.\d{1,2}.\d{2,4}` tem a mesma finalidade da expressão anterior, com a única diferença de utilizar ponto como separador, por exemplo: **15.12.1968**;

`\d{1,2}` e `\d{2,4}` são utilizadas no método LEGISLACAO onde representam dia e ano, respectivamente, sendo utilizadas em conjunto com termos sinalizadores de nomes de meses, visando a compor datas onde o mês aparece por extenso, por exemplo **15 dezembro 1968**, caso este que ocorre também na propriedade *Data* do conceito *Legislacao*.

Exemplo de tags resultantes consistem em exemplos do resultado de cada método, onde se demonstra, no formato de tags, as instâncias de propriedades e conceitos identificadas no documento, através da busca por características de localização, ocorrência e sinalização.

O apêndice B apresenta o resultado da extração a que foi submetido o documento contido no anexo A. Tomando-se apenas o primeiro parágrafo do documento, a título de exemplo, “APELAÇÃO CÍVEL Nº 999.999-9, DA PRIMEIRA VARA CÍVEL DA COMARCA DE APUCARANA” e submetendo-o ao módulo de extração, a lista de tags resultante seria:

```
1: <class:Acordao> 1
2:   <dataType:URL> 2
3:     C:\ONTO\corpus\acordao9999999.txt
4:   </dataType:URL>
5:   <object:temApelacao inverse=ehApelacaoDe> 3
6:     <class:Apelacao> 4
```

¹Conceito Raiz

²Propriedade Raiz de dados

³Propriedade Raiz de objeto

⁴Conceito Ramo


```

7:          <object:temComarca inverse=ehComarcaDe> 5
8:              <class:Comarca> 6
9:                  <dataType:Nome>
10:                     apucarana
11:                 </dataType:Nome>
12:             </class:Comarca>
13:         </object:temComarca>
14:     <object:temCompetencia>
15:         <class:Civel> 7
16:     </class:Civel>
17: </object:temCompetencia>
18: <dataType:Numero> 8
19:     999.999-9
20: </dataType:Numero>
21: <object:temEscrivania inverse=ehEscrivaniaDe>
22:     <class:Escrivania>
23:         <dataType:Nome>
24:             primeira vara cível
25:         </dataType:Nome>
26:     </class:Escrivania>
27: </object:temEscrivania>
28: </class:Apelacao>
29: </object:temApelacao>
30: </class:Acordao>

```

da qual pode-se obter a seguinte semântica:

Linha 1: o conceito raiz é do tipo *Acordao* – esta tag é fixa para todo documento, visto que a presente aplicação tem seu escopo limitada a documentos do tipo acórdão;

Linha 2: relacionada ao conceito *Acordao* há uma propriedade de dados *URL* – esta tag também é fixada em todo documento, visando a informar a localização do documento;

Linha 3: “C:\ONTO\corpus\acordao9999999.txt” é a instância da propriedade *URL* – a infor-

⁵Propriedade Ramo de objeto

⁶Conceito Relacionado

⁷Conceito Relacionado (sem indivíduo)

⁸Propriedade Ramo de dados

mação do indivíduo é derivada do arquivo que foi aberto;

Linha 5: relacionada ao conceito *Acordao* há uma propriedade de objeto *temApelacao*, cuja propriedade inversa se chama *ehApelacaoDe* – esta propriedade, bem como a sua inversa, foi inferida pelo método de extração RECURSO, a partir da linha 1 (localização) e do termo bonus “APELAÇÃO” (sinalização), visando a relacionar este subconceito de recurso ao corrente acórdão através de uma propriedade de relacionamento de objetos (object). A formação do nome desta propriedade, bem como de várias outras propriedades de objeto, se dá através da junção do prefixo “tem” ao termo encontrado capitalizado e subtraído de acentuações (*Apelacao*, no caso). Para facilitar o posterior processo de população, seguido do nome da propriedade vem o nome da propriedade inversa formada pelo termo “inverse=” seguido pelo prefixo “eh”, seguido do termo extraído e terminada pelo sufixo “De”;

Linha 6: há um conceito *Apelacao*, o qual se relaciona ao conceito *Acordao* através da propriedade *temApelacao* declarada na linha anterior – este conceito foi gerado pelo mesmo método da linha anterior, sendo formado também a partir do termo “APELAÇÃO”, capitalizado e subtraído de acentuações, antecedido do prefixo “class:”;

Linha 7: relacionada ao conceito *Apelacao* há uma propriedade de objeto *temComarca*, cuja propriedade inversa se chama *ehComarcaDe* – esta propriedade, bem como a sua inversa, foi inferida pelo método COMARCA, a partir da linha 1 e do termo bonus “COMARCA” (localização), visando a relacionar um conceito de comarca ao corrente conceito de recurso;

Linha 8: há um conceito *Comarca* relacionado ao conceito *Apelacao* através da propriedade *temComarca* – este conceito foi gerado pelo mesmo método da linha anterior, sendo formado também a partir do termo “COMARCA”, capitalizado e subtraído de acentuações, antecedido do prefixo “class:”;

Linha 9: relacionada ao conceito *Comarca* há uma propriedade de dados *Nome* – esta propriedade foi também gerada pelo método da linha anterior visando a associar um indivíduo ao conceito de comarca, através de uma tag formada pelo prefixo “dataType:” seguida pelo termo “Nome”;

Linha 10: “apucarana” é a instância da propriedade *Nome* – adicionada, por fim, pelo método COMARCA para incluir como propriedade do conceito de comarca o texto da linha 1 que vem após o bonus “COMARCA” (localização);

- Linha 14:** relacionada ao conceito *Apelacao* há uma propriedade de objeto *temCompetencia*, cuja propriedade inversa se chama *ehCompetenciaDe* – esta propriedade foi inferida pelo método de extração RECURSO, a partir da linha 1 (localização) e do termo bonus “CÍVEL” (sinalização), visando a relacionar um conceito de competência ao corrente conceito, ou subconceito, de recurso, através de uma propriedade de relacionamento de objetos formada pelo prefixo “object:” seguida pelo termo “temCompetencia”;
- Linha 15:** há um conceito *Civel* relacionado ao conceito *Apelacao* através da propriedade *temCompetencia* – adicionada pelo método RECURSO para incluir na lista um conceito de competência formada pelo próprio termo bonus “CÍVEL”, capitalizado e subtraído de acentos;
- Linha 18:** relacionada ao conceito *Apelacao* há uma propriedade de dados *Numero* – esta propriedade foi gerada pelo método RECURSO a partir da linha 1 (localização) e da validação de uma expressão regular (sinalização), conforme descrito no item a seguir, visando a associar um indivíduo ao corrente conceito, ou subconceito, de recurso, através de uma tag formada pelo prefixo “dataType:” seguida pelo termo “Numero”;
- Linha 19:** “999999-9” é a instância da propriedade *Numero* – adicionada pelo método RECURSO para incluir como propriedade do conceito de recurso o texto que se adéqua à expressão regular “[0-9[\u002E][\u002D][\u002F]]+” (sinalização), a qual busca um padrão composto por um ou mais algarismos intercalados pelos caracteres ‘.’, ‘-’ e ‘/’;
- Linha 21:** relacionada ao conceito *Apelacao* há uma propriedade de objeto *temEscrivania*, cuja propriedade inversa se chama *ehEscrivaniaDe* – esta propriedade, bem como a sua inversa, foi inferida pelo método ESCRIVANIA, a partir da linha 1 e do termo bonus “VARA” (localização), visando a relacionar um conceito de escrivania ao corrente conceito, ou subconceito, de recurso e cujo nome foi formado pela junção do prefixo “object:” seguida pelo termo “temEscrivania”;
- Linha 22:** há um conceito *Escrivania* relacionado ao conceito *Apelacao* através da propriedade *temEscrivania* – este conceito foi gerado pelo mesmo método da linha anterior, sendo formado também a partir do termo “VARA”, sendo nominado pelo prefixo “class:” seguido do termo “Escrivania”;
- Linha 23:** relacionada ao conceito *Escrivania* há uma propriedade de dados *Nome* – esta propriedade foi também gerada pelo método da linha anterior visando a associar um indivíduo ao conceito de escrivania, através de uma tag formada pelo prefixo “dataType:” seguida pelo termo “Nome”;

Linha 24: “primeira vara cível” é a instância da propriedade *Nome* – adicionada, por fim, pelo método *ESCRIVANIA* para incluir como propriedade do conceito de *escrivania* uma sentença composta da seguinte forma: o termo *bonus* encontrado (*VARA*); antecedido pelo termo que no texto vem antes, desde que se trate de um ordinal escrito por extenso (*PRIMEIRA*) ou numericamente (1^a, se fosse o caso); seguida da palavra que no texto vem depois do *bonus* (*CÍVEL*).

A estratégia de representar o resultado da extração neste formato visa a facilitar a varredura pelo processo de população seguinte, através de uma lista de formato simples, a *listTag*, evitando assim a complexidade de detalhes do RDF, tarefa esta atribuída ao Jena.

3.2.4 Módulo de População

Este módulo visa à população da ontologia com as instâncias de conceitos e propriedades extraídas do corpus. A invocação do módulo de população se dá dentro da mesma iteração onde ocorre a invocação do módulo de extração, significando que cada documento submetido à extração, tem as suas instâncias imediatamente inseridas no A-Box da ontologia residente na memória.

De maneira resumida, o algoritmo varre a lista de *tags* em memória, sendo que, para cada uma, invoca a biblioteca Jena para efetuar a sua inserção na ontologia, a qual também está na memória. Os passos efetuados pelo algoritmo foram descritos em pseudo-código, baseado na linguagem Pascal, e são os seguintes:

Algoritmo População da Ontologia JurisTJPR

Entrada: *listTag* {lista de tags resultante do processo de extração}

- 1: {processar o conceito raiz, i.e.:}
- 2: *item* := *listTag*.primeiroItem
- 3: **if**(*item* contém ‘ < class’)
- 4: **begin**
- 5: *tagConcRaiz* := extrair de *item* palavra após prefixo ‘class’
- 6: *concRaiz* := Jena buscar no TBox conceito = *tagConcRaiz*
- 7: *indivConcRaiz* := Jena criar no ABox *concRaiz*
- 8: **end**

```

9: while(item := listTag.proximoItem) do
10:  if(item contém ‘ < ’ AND NOT item contém ‘ / ’)
11:  begin
12:    if(item contém ‘object’ AND tagConcRamo = “)
13:    begin
14:      {então memorizar propriedade raiz a ser instanciada em iteração posterior, i.e.:}
15:      tagPropRaiz := extrair de item palavra após prefixo ‘object’
16:      invTagPropRaiz := extrair de item palavra após prefixo ‘inverse’
17:    end
18:  else
19:  begin
20:    if(item contém ‘class’)
21:    begin
22:      tagConcRamo := extrair de item palavra após prefixo ‘class’
23:      {processar o conceito ramo que vai se relacionar ao conceito raiz
24:       através da propriedade raiz de relacionamento, i.e.:}
25:      concRamo := Jena buscar no TBox conceito = tagConcRamo
26:      indivConcRamo := Jena criar no ABox concRamo
27:      propRaiz := Jena buscar no TBox propriedade = tagPropRaiz
28:      Jena criar no ABox propRaiz relacionando indivConcRaiz a indivConcRamo
29:      if(invTagPropRaiz <> “)
30:      begin
31:        invPropRaiz := Jena buscar no TBox propriedade = invTagPropRaiz
32:        Jena criar no ABox invPropRaiz relacionando indivConcRamo a indivConcRaiz
33:      end
34:    end
35:  else
36:  begin
37:    {memorizar propriedade ramo, i.e.:}
38:    tagPropRamo := extrair de item palavra após prefixo ‘object’
39:    invTagPropRamo := extrair de item palavra após prefixo ‘inverse’
40:    {pular próx. linha – deve haver conceito relacionado a ser memorizado, i.e.:}
41:    item := listTag.proximoItem
42:    tagConcRelacionado := extrair de item palavra após prefixo ‘class’

```

```

43:      {pular próx. linha – pode haver indivíduo para o conceito relacionado, i.e.:}
44:      item := listTag.proximoItem
45:      concRelacionado := Jena buscar no TBox conceito = tagConcRelacionado
46:      if(item contém ‘ < ’ AND NOT item contém ‘/’)
47:      begin
48:          indivConcRelacionado := Jena criar no ABox concRelacionado
49:          tagPropRelacionado := extrair de item palavra após prefixo ‘dataType’
50:          {pular próx. linha – para processar o individuo, i.e.:}
51:          item := listTag.proximoItem
52:          propRelacionado := Jena buscar no TBox propriedade = tagPropRelacionado
53:          Jena criar no ABox propRelacionado
                    atribuindo valor item a indivConcRelacionado
54:          propRamo := Jena buscar no TBox propriedade = tagPropRamo
55:          Jena criar no ABox propRamo
                    relacionando indivConcRamo a indivConcRelacionado
56:          invPropRamo := Jena buscar no TBox propriedade = invTagPropRamo
57:          Jena criar no ABox invPropRamo
                    relacionando indivConcRelacionado a indivConcRamo
58:          {pular 2 linhas – para saltar tags finais do conceito relacionado, i.e.:}
59:          listTag.proximoItem
60:          listTag.proximoItem
61:      end
62:      else
63:      begin
64:          {encontrou tag final – processar conceito relacionado sem indivíduo, i.e.:}
65:          propRamo := Jena buscar no TBox propriedade = tagPropRamo
66:          Jena criar no ABox propRamo relacionando indivConcRamo a concRelacionado
67:      end
68:      {pular 1 linha – para saltar a tag final da propriedade ramo, i.e.:}
69:      listTag.proximoItem
70:      end
71:  end
72: end

```

```

73:  if(item contém ‘ < /’)
74:    concRamo := NULL
75:  else
76:  begin
77:    if(itemAnterior contém ‘ < dataType’)
78:    begin
79:      if(concRamo <> NULL)
80:      begin
81:        tagPropRamo := extrair de item palavra após prefixo ‘dataType’
82:        propRamo := Jena buscar no TBox propriedade = tagPropRamo
83:        Jena criar no ABox propRamo atribuindo valor item ao indivConcRamo
84:      end
85:    else
86:    begin
87:      {processar indivíduo do conceito raiz, i.e.:}
88:      tagPropRaiz := extrair de item palavra após prefixo ‘dataType’
89:      propRaiz := Jena buscar no TBox propriedade = tagPropRaiz
90:      Jena criar no ABox propRaiz atribuindo valor item ao indivConcRaiz
91:    end
92:  end
93: end
94:  itemAnterior := item
95: end while

```

onde:

numerais à esquerda, seguidos de ‘:’, não consistem em instruções, figurando no algoritmo apenas como numeradores de cada linha do código;

termos em itálico representam variáveis ou identificadores;

sentenças entre chaves representam comentários que introduzem ou esclarecem as instruções subsequentes;

termos em negrito representam comandos e operadores lógicos

:= é um operador de atribuição de valor (à direita) a uma variável (à esquerda);

= é um operador lógico de igualdade entre dois membros;

<> é um operador lógico de diferença entre dois membros;

“ representa uma cadeia de caracteres vazias – quando precedido do operador de atribuição comanda o esvaziamento de uma variável de cadeia de caracteres; quando precedido de um operador lógico de igualdade ou diferença comanda a verificação do preenchimento de uma variável de cadeia de caracteres;

NULL quando precedido do operador de atribuição comanda o esvaziamento de uma variável; quando precedido de um operador lógico de igualdade ou diferença comanda a verificação do preenchimento de uma variável;

conceito raiz é o conceito principal, de maior nível, na hierarquia da listTag, subordinado ao qual estão todos os demais conceitos e propriedades. Este termo, assim como os abaixo descritos, faz parte de uma terminologia cunhada pelo autor visando a diferenciar os vários níveis hierárquicos da lista de tags, facilitando assim a especificação e o entendimento do algoritmo;

conceito ramo é um conceito subordinado ao *conceito raiz*, com o qual se relaciona através de uma *propriedade raiz* de objeto;

conceito relacionado é um conceito subordinado ao *conceito ramo*, com o qual se relaciona através de uma *propriedade ramo* de objeto;

propriedade raiz é uma propriedade do *conceito raiz*, podendo ser de objeto, visando ao relacionamento com *conceitos ramo*, ou de dados, visando à atribuição de um valor;

propriedade raiz inversa é a inversa de uma *propriedade raiz* de objeto, visando ao relacionamento entre um *conceito ramo* e o *conceito raiz* ao qual está subordinado;

propriedade ramo é uma propriedade do *conceito ramo*, podendo ser de objeto, visando ao relacionamento com *conceitos relacionados*, ou de dados, visando à atribuição de um valor;

propriedade ramo inversa é a inversa de uma *propriedade ramo* de objeto, visando ao relacionamento entre um *conceito relacionado* e o *conceito ramo* ao qual está subordinado.

A título de ilustração, pode-se executar o algoritmo de população tendo como entrada a lista de *tags* incluída na página 115, a qual por sua vez consiste em um trecho da seção B, página 166. Assim, no Quadro 9, apresentam-se as *tags* que foram percorridas por cada linha do algoritmo.

Linha do Algoritmo	Tag percorrida
2	<class:Acordao>
3	<class
5	Acordao
6	TBox ? Acordao
7	ABox + TBox.Acordao
9	<object:temApelacao inverse=ehApelacaoDe> <class:Apelacao> <object:temComarca inverse=ehComarcaDe> <object:temCompetencia> <dataType:Numero> 999999-9 <dataType:URL> C:\ONTO\corpus\acordao9999999.txt
12	object
15	temApelacao
16	ehApelacaoDe
20	class
22	Apelacao
25	TBox ? Apelacao
26	ABox + TBox.Apelacao
27	TBox ? temApelacao
28	ABox + TBox.temApelacao (ABox.Acordao, ABox.Apelacao)
31	TBox ? ehApelacaoDe
32	ABox + TBox.ehApelacaoDe (ABox.Apelacao, ABox.Acordao)
38	temComarca temCompetencia
39	ehComarcaDe
41	<class:Comarca> <class:Civel>
42	Comarca Civel

(continua)

Quadro 9: Exemplo de tags varridas pelo algoritmo de população

Linha do Algoritmo	Tag percorrida
44	<dataType:Nome> </class:Civel>
45	TBox ? Comarca TBox ? Civel
48	ABox + TBox.Comarca
49	Nome
51	apucarana
52	TBox ? Nome
53	ABox + TBox.Nome (ABox.Comarca, apucarana)
54	TBox ? temComarca
55	ABox + TBox.temComarca (ABox.Apelacao, ABox.Comarca)
56	TBox ? ehComarcaDe
57	ABox + TBox.ehComarcaDe (ABox.Comarca, ABox.Apelacao)
59	</dataType:Nome>
60	</class:Comarca>
65	TBox ? temCompetencia
66	ABox + TBox.temCompetencia (ABox.Apelacao, TBox.Civel)
69	</object:temComarca> </object:temCompetencia>
73	</class:Apelacao>
77	<dataType:Numero>
81	Numero
82	TBox ? Numero
83	ABox + TBox.Numero (ABox.Apelacao, 999999-9)
88	URL
89	TBox ? URL
90	ABox + TBox.URL (ABox.Acordao, C:\ONTO\corpus\acordao9999999.txt)

Quadro 9: Exemplo de tags varridas pelo algoritmo de população

onde:

Linha do Algoritmo indica o número da linha executada do algoritmo;

Tag percorrida corresponde à tag que foi avaliada pela correspondente linha do algoritmo. A presença de mais de uma tag para uma mesma linha, indica as tags que foram percorridas pela mesma linha em seguidas iterações do algoritmo;

TBox ? representa a busca da definição de um conceito ou propriedade no t-box;

ABox + representa a instanciação no a-box de um indivíduo correspondente a um conceito ou propriedade localizado no t-box;

TBox ? c indica que o algoritmo buscou o conceito c no t-box, por exemplo: *TBox ? Acordao* significa que o conceito *Acordao* foi procurado no t-box;

ABox + C indica que o algoritmo instanciou no a-box um indivíduo para o conceito C , encontrado no t-box, por exemplo: *ABox + TBox.Acordao* indica que o conceito *Acordao*, localizado no tbox, foi instanciado no a-box;

TBox ? p indica que o algoritmo buscou a propriedade p no t-box, por exemplo:

TBox ? temApelacao significa que a propriedade *temApelacao* foi procurada no t-box;

ABox + $P(I_1, I_2)$ indica que o algoritmo instanciou no a-box um indivíduo para a propriedade P , encontrada no t-box, correlacionando dois indivíduos I_1 e I_2 , instanciados anteriormente no a-box. *ABox + TBox.temApelacao (ABox.Acordao, ABox.Apelacao)*, por exemplo, indica que a propriedade *temApelacao*, localizada no t-box, foi instanciada no a-box para correlacionar os indivíduos de *Acordao* e *Apelacao*, já criados no a-box;

ABox + $P(I, v)$ indica que o algoritmo instanciou no a-box um indivíduo para a propriedade P , encontrada no t-box, atribuindo um valor v ao indivíduo I , instanciado anteriormente no a-box. Como exemplo: *ABox + TBox.Nome (ABox.Comarca, apucarana)* indica que a propriedade *Nome*, localizada no t-box, foi instanciada no a-box para atribuir o valor *apucarana* ao indivíduo de *Comarca*, já criado no a-box;

ABox + $P(I, C)$ indica que o algoritmo instanciou no a-box um indivíduo para a propriedade P , encontrada no t-box, correlacionando o indivíduo I , instanciado anteriormente no a-box, ao conceito C , encontrado no t-box e para o qual não há indivíduo. Como exemplo: *ABox + TBox.temCompetencia (ABox.Apelacao, TBox.Civel)* indica que a propriedade *temCompetencia*, localizada no t-box, foi instanciada no a-box para correlacionar o indivíduo de *Apelacao*, já criado no a-box, ao conceito *Civel*, localizado no t-box;

Assim, após a varredura de todas as tags da *listTag*, todos os indivíduos inseridos na ontologia residente em memória, poderão ser visualizados voltando-se à janela do módulo de visualização, conforme a Figura 23.

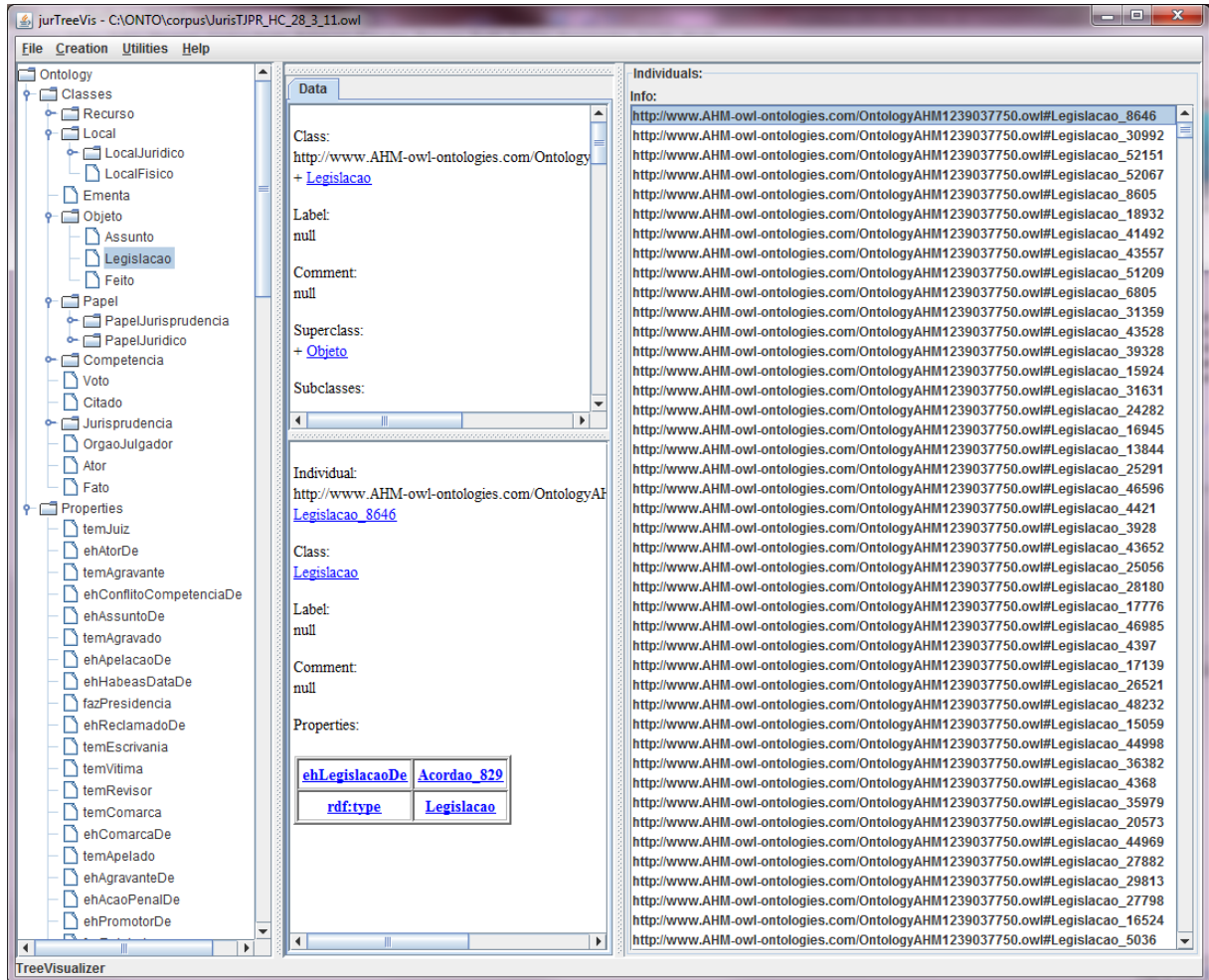


Figura 23: Visualização dos indivíduos populados na ontologia

onde:

- o quadro da direita, intitulado *Individuals* (indivíduos), apresenta uma lista contendo os indivíduos pertencentes ao conceito correntemente selecionado na árvore hierárquica (quadro da esquerda);
- o quadro inferior central apresenta detalhes do indivíduo correntemente selecionado: URI do indivíduo (*Individual*), classe à qual pertence (*Class*), rótulo (*Label*), comentários (*Comment*) e propriedades (*Properties*), consistindo em um quadro onde a primeira coluna mostra o nome da propriedade e a segunda o objeto relacionado;

Por fim, nesta janela pode-se gravar a ontologia populada permanentemente, onde uma

chamada à biblioteca Jena procede à gravação no formato OWL-DL, resultando em um arquivo que contém tanto o T-Box quanto o A-Box da ontologia.

3.2.5 Módulo de Pesquisa

O último módulo visa à busca de sentenças, a serem especificadas por um usuário, nos acórdãos que compõe o corpus. Em uma busca tradicional, isto se faria de maneira textual, ou seja, se buscaria cada sentença, indistintamente, através do conteúdo completo de cada documento. O presente trabalho almeja uma **busca semântica**, então a pesquisa aqui se dará pela ontologia, através da procura por indivíduos de conceitos (selecionados pelo usuário) que contenham as sentenças desejadas. Como cada indivíduo, independentemente do conceito a que pertença, sempre estará relacionado a um indivíduo do conceito *Acórdão*, a ligação entre eles levará ao caminho do documento, através da sua propriedade URL, compondo assim o resultado da pesquisa. Para ilustrar isso, tome-se um trecho da taxonomia resultante do terceiro refinamento (Figura 17 da página 90), conforme a Figura 24 abaixo.

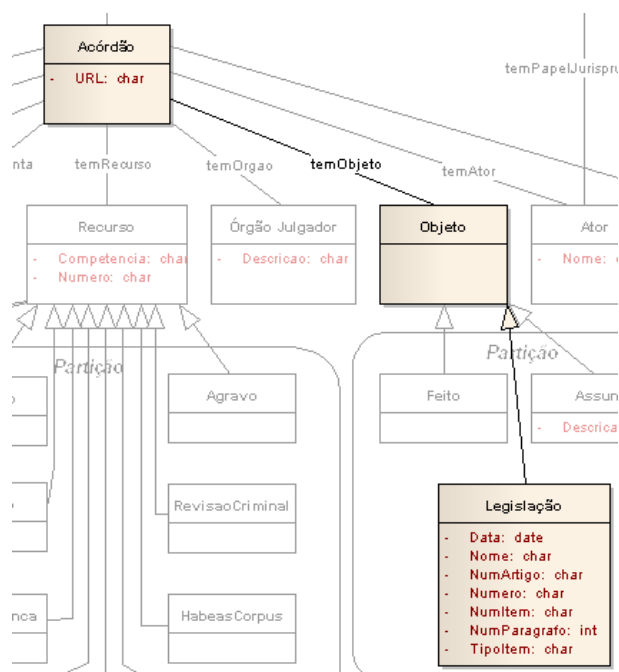


Figura 24: Relacionamento Legislação/Acórdão na taxonomia

Nota-se na figura que o conceito *Legislacao*, por ser uma especialização do conceito *Objeto*, tem um relacionamento com o conceito *Acórdão*, através da propriedade *temObjeto*. Por conseguinte, um indivíduo do conceito *Legislacao* estará ligado a um indivíduo do conceito *Acórdão*, o qual terá a sua propriedade *URL* preenchida por um literal que indica o caminho e o nome do documento que contém o acórdão em questão.

Naturalmente que, para haver algum resultado, a ontologia deve possuir um A-Box, além do T-Box, ou seja, deve ter sido populada em algum momento anterior à pesquisa. Pode ser imediatamente antes – caso o processo de população tenha acabado de acontecer – ou em algum momento antes – caso a ontologia tenha sido populada e gravada em arquivo, através do módulo de visualização, visando a posteriores pesquisas. No segundo caso, o arquivo deve ser aberto (também através do módulo de visualização), tendo como consequência a carga da ontologia em memória. Desta maneira é possível, na janela **Pesquisa**, formular e submeter consultas à ontologia, visando a obter informações resultantes dos documentos que contém indivíduos que se adequam à consulta executada sobre a ontologia, conforme demonstrado na Figura 25,

The screenshot shows the 'Pesquisa' window with the following elements:

- Search Form:**
 - Text input: 8069
 - Concept dropdown: Legislacao
 - Property dropdown: Numero
 - Logic: E, OU
 - Match type: Idêntico
 - Text area: E:Legislacao:Numero=11340, E:Legislacao:Numero=8069
 - Buttons: Remover, Distinct, Pesquisar
- Results Table:**

raiz1	URLraiz	val_Numero0	val_Numero1
onto:Acordao_2230	"C:\ONTO\lcorpus\ACORDAO_813177_1078821.TXT"	"11340/06"	"8069/1990"
onto:Acordao_2230	"C:\ONTO\lcorpus\ACORDAO_813177_1078821.TXT"	"11340/06"	"8069/90"
onto:Acordao_2230	"C:\ONTO\lcorpus\ACORDAO_813177_1078821.TXT"	"11340/06"	"8069/1990"
onto:Acordao_2233	"C:\ONTO\lcorpus\ACORDAO_813180_1079368.TXT"	"11340/06"	"8069/90"
onto:Acordao_2233	"C:\ONTO\lcorpus\ACORDAO_813180_1079368.TXT"	"11340/06"	"8069/1990"
onto:Acordao_2233	"C:\ONTO\lcorpus\ACORDAO_813180_1079368.TXT"	"11340/06"	"8069/90"
onto:Acordao_2233	"C:\ONTO\lcorpus\ACORDAO_813180_1079368.TXT"	"11340/2006"	"8069/1990"
onto:Acordao_3237	"C:\ONTO\lcorpus\ACORDAO_835802_1079403.TXT"	"11340/06"	"8069/90"
onto:Acordao_3237	"C:\ONTO\lcorpus\ACORDAO_835802_1079403.TXT"	"11340/06"	"8069/90"
onto:Acordao_3991	"C:\ONTO\lcorpus\ACORDAO_855649_1079193.TXT"	"11340/06"	"8069/1990"
onto:Acordao_3991	"C:\ONTO\lcorpus\ACORDAO_855649_1079193.TXT"	"11340/06"	"8069/90"
onto:Acordao_3991	"C:\ONTO\lcorpus\ACORDAO_855649_1079193.TXT"	"11340/06"	"8069/1990"
onto:Acordao_3991	"C:\ONTO\lcorpus\ACORDAO_855649_1079193.TXT"	"11340/06"	"8069/90"
- SPARQL Query:**

```

PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>
PREFIX onto: <http://www.AHM-owl-ontologies.com/OntologyAHM1239037750.owl#>
PREFIX rdf: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?raiz0 ?raiz1 ?URLraiz ?val_Numero0 ?val_Numero1
WHERE
{
  ?indivuo0 onto:Numero ?val_Numero0 ;
  onto:ehLegislacaoDe ?raiz0 .
  ?raiz0 onto:URL ?URLraiz .
  ?indivuo1 onto:Numero ?val_Numero1 ;
  onto:ehLegislacaoDe ?raiz1 .
  ?raiz1 onto:URL ?URLraiz .
  FILTER ((( regex(afn:localname(?indivuo0), "Legislacao", "I") && regex(?val_Numero0, "11340", "I") )
  && ( regex(afn:localname(?indivuo1), "Legislacao", "I") && regex(?val_Numero1, "8069", "I") )
  && ( ?raiz0 = ?raiz1 ) ) )
}
ORDER BY ?URLraiz

```

Figura 25: jurTreeVis – janela de pesquisa

onde:

- está exemplificada uma busca por acórdãos onde se abordem ambas as leis do Estatuto da Criança e do Adolescente (ECA, nº 8.069) e a Maria da Penha (nº 11.340)
- a caixa de texto no canto superior esquerdo da janela deve ser preenchida com a sentença procurada pelo usuário. Na figura está se montando a consulta para a legislação referente

ao ECA pelo respectivo número, por isso a caixa foi preenchida com o termo “8069”. Se fosse desejável a busca pelo nome da lei, poderia ter-se informado a respectiva sentença, completa ou parcialmente;

- a caixa intitulada **Conceito** contém uma lista de conceitos existentes na ontologia, sendo que um deles deve ser selecionado para que a busca se restrinja a indivíduos exclusivamente subsumidos por este conceito. A lista foi fixada na aplicação, consistindo nos seguintes conceitos: *Ator*, *Assunto*, *Comarca*, *Escrivanía*, *Fato*, *Legislação*, *Órgão Julgador*, *Recurso*, *Relator* e *Voto* – os quais consistem naqueles de maior interesse para uma pesquisa jurisprudencial. No exemplo, foi selecionado o conceito *Legislação* visto que a consulta visa a busca por lei;
- a caixa intitulada **Propriedade** contém uma lista das propriedades relativas ao conceito selecionado na lista anterior. Assim, para os conceitos *Ator*, *Comarca*, *Escrivanía* e *Relator* é disponibilizada a propriedade *Nome*. Para os conceitos *Assunto*, *Fato*, *Orgao-Julgador* e *Voto* é disponibilizada a propriedade *Descrição*. Para o conceito *Recurso* é disponibilizada a propriedade *Numero*. E, por fim, para o conceito *Legislação* são disponibilizadas as propriedades *Nome*, *TipoItem*, *NumParagrafo*, *Numero*, *NumItem*, *Tipo*, *NumArtigo* e *Data*. Desta maneira, a busca será restrita, não somente pelo conceito, mas também por uma de suas propriedades. No exemplo foi selecionada a propriedade *Numero* visto que se pretende pesquisar a lei através de seu número;
- no agrupador **E/OU**, logo abaixo do componente anterior, deve ser selecionado um dos conectores para indicar qual tipo de conexão lógica se deseja entre as várias buscas que se pretende submeter à ontologia;
- o botão **Adicionar** deve ser clicado para incluir cada expressão de busca, especificada através de todos os componentes anteriores, na lista da pesquisa;
- a caixa abaixo do botão Adicionar, contém a lista das expressões de busca especificadas pelo usuário a qual é posteriormente transformada em uma consulta SPARQL a ser submetida à ontologia em memória. Cada expressão adicionada à caixa baseia-se na seguinte sintaxe (definida pelo autor):

$$C : c : p = s(I)$$

onde:

C indica o *conector lógico* selecionado para a expressão, E ou OU (o corrente aplicativo, por uma questão de complexidade de implementação, replica o conector da primeira expressão para todas as subsequentes, independentemente daquele que se selecionou em cada uma);

c indica o *conceito* que foi selecionado;

p indica a *propriedade* relativa ao conceito que foi selecionada;

s indica a *sentença* especificada na primeira caixa de texto;

(I) é um indicador de que os literais procurados devem ser idênticos à sentença especificada pelo usuário.

Assim, no exemplo da Figura 25 a expressão “**E:Legislacao:Numero:11340**” especifica que deve-se buscar um indivíduo subsumido pelo conceito *Legislacao*, cuja propriedade *Numero* o relaciona a um objeto literal que contém o termo *11340*. A segunda expressão, “**E:Legislacao:Numero:8069**”, terá o mesmo significado que a anterior, variando apenas no literal que deverá conter *8069*. O conector E das expressões especifica que deverá haver uma intersecção entre os resultados obtidos pelas expressões da lista, ou seja, os indivíduos que contém os literais 11340 e 8069 devem se relacionar a um mesmo indivíduo do conceito *Acordao*;

- o botão **Remove** possibilita a remoção de expressões da lista;
- o botão **Pesquisar** gera a consulta SPARQL e submete-a à ontologia visando à busca por triplas que atendam às expressões da lista. Assim, no exemplo da Figura 25, a pesquisa resultará em indivíduos do conceito *Acordao* os quais se relacionem a pelo menos dois indivíduos do conceito *Legislacao*, cujos literais da propriedade *Numero* sejam 11340 e 8069;
- a caixa de texto na região inferior da janela visa à saída da consulta SPARQL, construída para a pesquisa na ontologia, baseada na lista de expressões. A finalidade desta saída, a princípio, é a mera visualização do artefato gerado;
- a caixa **Resultado** visa à saída das variáveis retornadas da submissão da consulta SPARQL sobre a ontologia em memória, as quais indicam os indivíduos do conceito *Acordao* encontrados, bem como os literais relacionados que contém os literais especificados nas expressões da lista. Por exemplo, na Figura 25 nota-se que a consulta encontrou o indivíduo **onto:Acordao.2230** cuja propriedade *URL* aponta para o literal

“C:\\ONTO\\corpus\\ACORDAO_813177_1078821.TXT” e que se relaciona a indivíduos de Legislação que contém as literais especificadas. As repetições de indivíduos de *Acordao* no resultado, devem-se àqueles que possuem mais de um indivíduo relacionado que atendem às expressões da lista. O cabeçalho existente no resultado reproduz os nomes das variáveis da *select list* da consulta;

- o botão **Reexecuta** é uma alternativa ao botão Pesquisar, que visa a submeter novamente a consulta SPARQL à ontologia, sem reconstruí-la. Isto é útil em caso de testes, em que se tenha alterado manualmente a consulta na caixa de textos inferior.

Há, ainda, mais componentes na janela de pesquisa que podem modificar o resultado da busca. Para analisá-los, tome-se como exemplo a Figura 26.

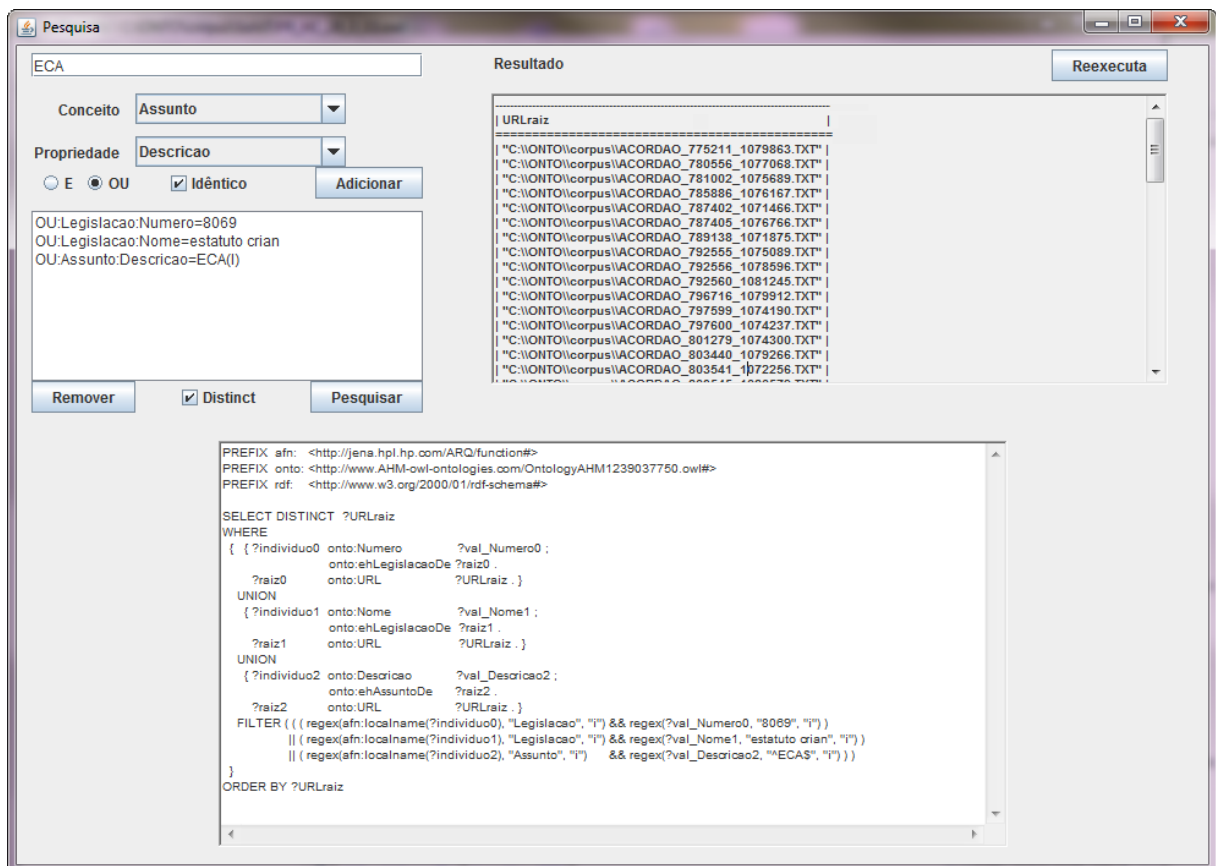


Figura 26: jurTreeVis – componentes adicionais da janela de pesquisa

onde:

- o marcador intitulado **Idêntico** modifica a busca para que o retorno ocorra somente para literais equivalentes por inteiro com a string de busca, fornecida na caixa superior. No exemplo, a busca terá sucesso para literais idênticos à sigla ECA. Já para literais que

contenham a sigla, além de outros caracteres, como por exemplo “*sueca*”, não haverá retorno;

- o marcador intitulado **Distinct** provoca a adição de uma cláusula *DISTINCT* à *select list* da consulta SPARQL a ser gerada, bem como a redução das variáveis que a compõe a apenas uma: *URL*. Com isso o resultado será obtido de forma mais concisa, não apresentando repetição de indivíduos do conceito *Acordao* e trazendo apenas a informação que efetivamente interessa para a pesquisa, a *URL* do documento que contém o texto integral do acórdão;

Assim, no exemplo da Figura 26 a expressão “**OU:Legislacao:Numero:8069**” especifica que deve-se buscar um indivíduo subsumido pelo conceito *Legislacao*, cuja propriedade *Numero* o relaciona a um objeto literal que contém o termo *8069*. A segunda expressão, “**OU:Legislacao:Nome:estatuto crian**”, requisita a busca de um indivíduo do mesmo conceito, porém com uma propriedade *Nome* relacionada a um literal que contenha a sentença “estatuto crian”. A última expressão, “**OU:Assunto:Descricao:ECA(I)**”, especifica que deve-se buscar um indivíduo subsumido pelo conceito *Assunto*, cuja propriedade *Descricao* o relaciona a um objeto literal igual ao termo “ECA” (a busca se dará por literais idênticos ao termo especificado graças à presença do indicador “(I)”). Relacionando todas as expressões, está o conector **OU**, o qual especifica que deverá haver uma união entre os resultados obtidos pelas expressões da lista, ou seja, um indivíduo do conceito *Acordao*, para ser incluído no resultado, deverá possuir pelo menos um dos literais procurados.

Quando o botão *Pesquisar* é clicado o presente módulo cria uma consulta baseada na sintaxe do SPARQL que atenda às expressões de pesquisa da lista. Tomando-se o segundo exemplo tem-se o código a seguir:

```

PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>
PREFIX onto: <http://www.AHM-owl-ontologies.com/OntologyAHM1239037750.owl#>
PREFIX rdf: <http://w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?URLRaiz
WHERE
  {
    { ?individuo0 onto:Numero ?val_Numero0 .
      ?individuo0 onto:ehLegislacaoDe ?raiz0 .
      ?raiz0 onto:URL ?URLRaiz . }
    UNION
    { ?individuo1 onto:Nome ?val_Nome1 .
      ?individuo1 onto:ehLegislacaoDe ?raiz1 .
      ?raiz1 onto:URL ?URLRaiz . }
    UNION
    { ?individuo2 onto:Descricao ?val_Descricao2 .
      ?individuo2 onto:ehAssuntoDe ?raiz2 .
      ?raiz2 onto:URL ?URLRaiz . }
    FILTER ((( regex(afn:localname(?individuo0), "Legislacao'", "i")
      && regex(?val_Numero0, "8069", "i"))
      || ( regex(afn:localname(?individuo1), "Legislacao'", "i")
      && regex(?val_Nome1, "estatuto crian", "i"))
      || ( regex(afn:localname(?individuo2), "Assunto'", "i")
      && regex(?val_Descricao2, " ^ECA$", "i")))
  }
ORDER BY ?URLRaiz

```

onde:

afn é um prefixo que aponta para um esquema de definições de funções para o SPARQL, disponibilizado pela framework Jena;

onto é um prefixo que resume as URIs das definições da ontologia jurisTJPR;

rdf é um prefixo que aponta para o esquema de definições da W3C RDF-Schema;

URLRaiz é uma variável onde se pretendem obter os literais da propriedade URL do conceito

dos acórdãos que atendem aos requisitos da consulta;

individuo0, 1 e 2 são variáveis que conterão as URIs dos conceitos que definem os indivíduos pesquisados;

Numero, Nome e Descricao são propriedades de dados definidas no T-Box (onde se encontram prefixadas pela URI que aqui é representada pelo prefixo onto) que se desejam encontrar nos conceitos representados pelas variáveis **individuo0, 1 e 2**, respectivamente;

val_Numero0, val_Nome1 e val_Descricao2 são variáveis que conterão os literais associados às propriedades **Numero, Nome e Descricao**, respectivamente;

ehLegislacaoDe e ehAssuntoDe são propriedades relacionais definidas no T-Box que visam a efetuar a relação entre os indivíduos (representados pelas variáveis **individuo0, 1 e 2**) e os acórdãos a que pertencem (identificados pelas variáveis **raiz0, 1 e 2**);

URL são propriedades de dados do conceito **Acordao**, definidas no T-Box;

Legislacao e Assunto são os nomes dos conceitos, especificados pelo usuário, que na consulta estão na cláusula **FILTER** para restringir que os indivíduos (representados pelas variáveis **individuo0, 1 e 2**) necessariamente pertençam a estes conceitos;

“8089”, “estatuto crian” e “^ECA\$” são as sentenças especificadas pelo usuário e que na cláusula **FILTER** estão restringindo que os literais dos indivíduos encontrados devem conter os termos em questão. A última sentença tem um diferencial, que visa a atender ao requisito *Idêntico* especificado pelo usuário: os caracteres **^** e **\$**, da sintaxe das expressões regulares, que restringem o início e o final, respectivamente, do literal.

A consulta é então submetida à interface da Jena *QueryFactory*, o qual transforma a string contendo o comando em um objeto da classe *Query*, que torna o comando interpretável pela Jena. O objeto *Query* é então submetido a outra interface da Jena, a *QueryExecutionFactory*, que cria um objeto da classe *QueryExecution* que consiste em um plano de execução para a consulta. Agora, basta executar o método *execSelect*, pertencente à classe *QueryExecution*, para se efetuar finalmente a busca, cujo resultado retorna em um objeto da classe *ResultSet*. Para apresentar o resultado formatado, é invocada mais uma interface, a *ResultSetFormatter*, à qual se submete o objeto *ResultSet*, resultando em uma listagem formatada do resultado o qual é, assim, atribuída à caixa *Resultado*.

3.3 AVALIAÇÃO DA ONTOLOGIA COM ENFOQUE NO USUÁRIO

O módulo de população, componente da aplicação, conforme descrito na seção 3.2.4, página 119, além de popular o modelo de ontologia residente em memória, produz um arquivo resultante para cada acórdão processado, visando a avaliação pelo usuário dos conceitos e propriedades populados, conforme pode-se ver no apêndice C.

Esta avaliação contou com o acesso a uma base de dados de testes do sistema de jurisprudência do TJPR, na qual foram encontrados 3.309 documentos de acórdãos, publicados pelos desembargadores daquele tribunal entre os anos 2002 e 2005.

3.3.1 Método da avaliação

Um aspecto determinante nesta avaliação de acórdãos populados na ontologia foi a inexistência de uma base de dados rotulados para aprendizagem prévia de máquina. Além disso, a modelagem de domínios, através de ontologias, geralmente tem forte dependência de especialistas. Estas características, juntas, tornaram a avaliação com enfoque no **usuário** o principal instrumento de validação da aplicação baseada em ontologia aqui desenvolvida.

No caso do presente trabalho, a avaliação teve de levar em consideração a escassez de recursos humanos na Seção de Jurisprudência, bem como o fato de que uma avaliação da consistência das instâncias dos conceitos extraídos de cada documento é muito custosa. Devido a estes fatos, foi definido que a avaliação devia se dar em uma amostra de tamanho viável para um ou dois especialistas e suficiente para se chegar a conclusões sobre eficácia e precisão da aplicação aqui postulada. Assim, foi definida uma quantidade de cinquenta documentos, de extensões variadas a serem sorteados dentre os 1.149 referentes ao ano de 2005.

A seleção da amostra foi aleatória, mas visando a homogeneidade em termos de extensão de documentos, ou seja o sorteio deveria manter a equitatividade no tamanho dos documentos sorteados. Assim, considerando que os tamanhos dos arquivos variam de 9 a 32 *Kbytes* foram definidas 23 faixas discretizadas, dentro das quais foram efetuados sorteios em quantidades homogêneas.

Cada acórdão da amostra foi processado pelo jurTreevis e o resultado da população – formatado em uma lista de indivíduos conforme exemplo do apêndice C – apresentado ao usuário especialista, junto com o texto integral do documento – conforme exemplo do anexo A, exceto pelas notas de rodapé.

Munidos das listas, resultantes da população, bem como do texto do acórdão, os especi-

alistas foram convidados a efetuar uma comparação entre a lista e o texto de cada acórdão, visando apontar indivíduos que eles julgaram tecnicamente inadequados para aquele acórdão, bem como sugerir outros indivíduos que deveriam, no seu julgamento, ter sido populados. O resultado desta avaliação foi tabulado, sendo apresentado e descrito no capítulo 4, página 142.

3.4 AVALIAÇÃO DA PESQUISA SEMÂNTICA

Pode-se considerar que o módulo de Pesquisa concentra todos os objetivos e expectativas do presente trabalho. A sua avaliação pode ser representativa do nível de sucesso alcançado neste projeto, bem como se ele pode proporcionar a melhora preconizada às pesquisas jurisprudenciais. Assim, pretende-se aqui avaliar o resultado obtido no módulo de pesquisa em buscas por temas relevantes na jurisprudência, através da análise de revocação e precisão.

O método para avaliação dos resultados obtidos no módulo de pesquisa precisou levar em consideração algumas restrições importantes:

- a base utilizada na avaliação com enfoque no usuário, conforme descrito na seção anterior, é muito reduzida, sendo insuficiente para uma análise segura;
- a *framework* do Jena, influenciada pelos recursos de memória do computador utilizado, apresentou limitação para as inserções no A-Box. Em geral não foi possível ultrapassar muito a quantidade de cinco mil acórdãos populados na ontologia;
- a base de dados de produção do sistema de jurisprudência do TJPR, contém dezenas de milhares de acórdãos por ano. Apesar da grande extensão da base, a pesquisa atual TJPR não possibilita restrição de período inferior a um ano;
- a quantidade de especialistas em jurisprudência disponíveis no TJPR é pequena.

Para transpor o primeiro obstáculo foi obtida uma nova cópia da base de dados de produção, atualizada no início do ano de 2010. Nesta foram encontradas as seguintes quantidades de acórdãos: 38.344 em 2005, 61.727 em 2006, 67.998 em 2007, 80.373 em 2008, 56.489 em 2009 e 1.580 em 2010. São quantidades mais que suficientes para a avaliação, porém excessivamente grandes perante a limitação do Jena, considerada na segunda restrição. Para compor um corpus suficiente, é necessária a definição de um corte na base de dados que resulte em uma quantidade viável para o experimento, posto que cinco mil é pouco, mas oitenta mil é excessivo.

Na definição do corte, a terceira restrição traz mais um empecilho: a pesquisa do TJPR permite limitação na base pesquisada baseado na unidade de ano – o que, já é sabido, ultrapassa

muito a capacidade aqui percebida para o Jena (cinco mil acórdãos). Entretanto, mesmo que fosse possível a definição de um período inferior (baseado em mês, por exemplo), surge uma quarta restrição: o método de avaliação de revocação, como foi visto na seção 2.5 (página 73), depende da definição de um conjunto de dados relevantes (ou esperados) o que torna-se humanamente inviável de se definir se considerados a necessidade de manipular vários milhares de documentos e a escassez de especialistas disponíveis. Um conjunto que poderia ser definido para este fim é o resultado obtido na pesquisa atual do TJPR. Entretanto a própria revocação dos resultados desta pesquisa não é conhecida, logo a sua utilização como conjunto de dados relevantes não seria confiável. Assim, dadas as restrições, o método de avaliação deverá partir de duas definições:

Extensão do corpus: quantidade de documentos que seja suficientemente representativa do domínio da jurisprudência, mas que não extrapole as capacidades da plataforma onde está inserido o aplicativo;

Conjunto de dados relevantes: para cada pesquisa avaliada, deve-se definir um conjunto de dados relevantes, ou seja, quais documentos são esperados no resultado.

3.4.1 Método da avaliação

Na definição da *extensão do corpus* considerou-se que a quantidade de acórdãos suportada pelo Jena, cinco mil, é insuficiente. Para solucionar este impasse, optou-se por compor vários corpus, limitados em cinco mil, a serem processados um de cada vez, em instâncias diferentes do aplicativo. A cada instância as mesmas pesquisas foram executadas, visando uma tabulação homogênea ao final. Assim, para ter-se uma quantidade suficiente e viável de documentos, foram definidos três corpus, totalizando assim quinze mil documentos.

Com estas definições, foram efetuadas três extrações de corpus, conforme descrito na subseção 3.2.1 da página 103, sendo os três corpus submetidos aos módulos de extração e população, resultando em três A-Box diferentes, sendo cada um gravado em um arquivo OWL junto com o respectivo T-Box. Na sequência, em três instâncias diferentes do aplicativo foram efetuadas as mesmas pesquisas semânticas, ou seja, cada pesquisa foi submetida às três ontologias populadas (as pesquisas efetuadas estão listadas na Tabela 12 do Capítulo 4).

Para se obter o *conjunto de documentos relevantes* esperados a cada pesquisa realizada, optou-se por compor uma base de dados relacional, onde cada corpus é representado por uma relação (tabela), sendo cada documento armazenado em um meta-dado (além dos dados dos respectivos números de processo e acórdão). Assim, para cada pesquisa semântica, foi efetu-

ada uma consulta SQL nas três relações, onde buscou-se exaurir, em suas cláusulas WHERE, todas as possibilidades de localização das sentenças pesquisadas no aplicativo. As tuplas (registros) resultantes então são consideradas como o conjunto de documentos relevantes esperados. Assim, por exemplo, para uma pesquisa semântica efetuada visando buscar acórdãos com referências à Lei Maria da Penha, através das seguintes expressões:

```
OU:Legislacao:Nome:penha
OU:Legislacao:Numero:11340
```

submeteu-se a seguinte consulta SQL às relações:

```
select numAcordao || '-' || processo as acordaoProcesso from acordao
where upper(texto) like '%LEI MARIA DA PENHA%'
OR upper(texto) like '%LEI MARIA PENHA%'
OR upper(texto) like '%LEI Ma PENHA%'
OR upper(texto) like '%LEI Ma. PENHA%'
OR upper(texto) like '%11.340/06%' OR upper(texto) like '%11.340-06%'
OR upper(texto) like '%11340/06%' OR upper(texto) like '%11340-06%'
OR upper(texto) like '%11.340/2006%' OR upper(texto) like '%11.340-2006%'
OR upper(texto) like '%11340/2006%' OR upper(texto) like '%11340-2006%'
OR upper(texto) like '%11.340 DE 2006%' OR upper(texto) like '%11340 DE 2006%'
OR upper(texto) like '%11.340 DE AGOSTO DE 2006%'
OR upper(texto) like '%11340 DE AGOSTO DE 2006%'
OR upper(texto) like '%11.340 DE 7 DE AGOSTO DE 2006%'
OR upper(texto) like '%11340 DE 7 DE AGOSTO DE 2006%'
OR upper(texto) like '%11.340 DE 07 DE AGOSTO DE 2006%'
OR upper(texto) like '%11340 DE 07 DE AGOSTO DE 2006%'
OR upper(texto) like '%11.340 DE 7/8/06%'
OR upper(texto) like '%11.340 DE 7/08/06%'
OR upper(texto) like '%11.340 DE 07/08/06%'
OR upper(texto) like '%11.340 DE 07/08/2006%'
OR upper(texto) like '%11340 DE 7/8/06%'
OR upper(texto) like '%11340 DE 7/08/06%'
OR upper(texto) like '%11340 DE 07/08/06%'
OR upper(texto) like '%11340 DE 07/08/2006%'
order by acordaoProcesso
```


onde:

numAcordao e processo são metadados dos números de acórdão e de processo original a que se referem o documento (dados extraídos da base de dados original);

acordaoProcesso é um *alias* que representa a concatenação dos dois metadados anteriores, visando a posterior comparação com o sufixo dos nomes dos arquivos do corpus, o qual é composto pelos dois metadados;

texto é um metadado que armazena o conteúdo do documento do acórdão;

Tabulando o resultado das duas consultas (semântica e relacional) submetidas ao primeiro corpus, chega-se à Tabela 10.

Tabela 10: Resultados das consultas à Lei Maria da Penha		
Pesquisa relação	Pesquisa semântica	C
acordaoprocesso	URLraiz	
797677_1076433	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_797677_1076433.TXT"	1
813177_1078821	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_813177_1078821.TXT"	1
813178_1079120	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_813178_1079120.TXT"	1
813179_1079213	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_813179_1079213.TXT"	1
813180_1079368	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_813180_1079368.TXT"	1
828839_1073665	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_828839_1073665.TXT"	1
828842_1076202	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_828842_1076202.TXT"	1
835422_1078112	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_835422_1078112.TXT"	1
835801_1074773	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_835801_1074773.TXT"	1
835802_1079403	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_835802_1079403.TXT"	1
845075_1078805	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_845075_1078805.TXT"	1
845076_1079137	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_845076_1079137.TXT"	1
848192_1081096	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_848192_1081096.TXT"	1
855649_1079193	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_855649_1079193.TXT"	1
917222_1078834	"C:\\\\ONTO\\\\corpus\\\\ACORDAO_917222_1078834.TXT"	1
14	14	14
	Revocação	100,00
	Precisão	100,00

Onde:

Pesquisa relação é a coluna onde se apresenta o resultado obtido na submissão da consulta SQL à relação, o qual é aqui considerado como o conjunto de documentos relevantes, ou seja o que se espera que a consulta semântica retorne;

Pesquisa semântica é a coluna onde se apresenta o resultado obtido da submissão da consulta à ontologia;

C é a coluna onde se apresenta o resultado da verificação da *consistência* do resultado da pesquisa semântica, o qual se deu da seguinte forma: o sufixo (numérico) do nome de arquivo contido em cada URL retornada deve ser idêntico à cada tupla retornada pela pesquisa relacional, considerando que ambas consultas incluem ordenação alfabética de seus resultados. Quando a verificação resultar em verdade, $C = 1$. Do contrário, $C = 0$. Na pesquisa ora avaliada, os resultados todos tiveram consistência verdadeira, significando que o resultado da consulta na ontologia se igualou totalmente ao resultado esperado;

14: a linha abaixo dos resultados apresenta os totais, sendo que na primeira célula está a quantidade total de tuplas que compõem o conjunto de dados esperado, a segunda o total de URLs retornadas (independentemente de sua consistência) e na última a soma dos valores da consistência de cada resultado obtido na consulta à ontologia;

Revocação expressa a taxa percentual de revocação alcançada pelo resultado da pesquisa na ontologia, em comparação ao conjunto de dados relevantes, calculada através da equação:

$$C = \frac{c}{r} \cdot 100$$

onde:

C representa a taxa de **Revocação** calculada;

c é o **somatório da consistência** de todos os resultados da pesquisa na ontologia;

r é a **quantidade** de tuplas do conjunto **de dados relevantes**;

Precisão expressa a taxa percentual de precisão alcançada pelo resultado da pesquisa na ontologia, ou seja a proporção do resultado que está correto, calculada através da equação:

$$P = \frac{c}{o} \cdot 100$$

onde:

P representa a taxa de **Precisão** calculada;

c é o **somatório da consistência** de todos os resultados da pesquisa na ontologia;

o é a **quantidade** de resultados obtidos na consulta à **ontologia**;

4 RESULTADOS

Neste capítulo pretende-se demonstrar os resultados obtidos na aplicação dos módulos de população e pesquisa, através da apresentação de relatos de duas avaliações: a primeira, com enfoque na satisfação dos usuários, visa a aferir a qualidade dos indivíduos extraídos e a segunda visa a conferir a qualidade dos resultados obtidos no módulo de pesquisa semântica.

4.1 AVALIAÇÃO DA EXTRAÇÃO DE INDIVÍDUOS

Três especialistas do TJPR se voluntariaram a participar da avaliação, conforme descrita na seção 3.3, página 136, ao longo de dois meses nos intervalos de suas atividades normais do TJPR, sendo que o resultado do seu trabalho foi tabulado conforme a Tabela 11.

Tabela 11: Resultado da avaliação com enfoque no usuário

(continua)

Acórdão	Indivíduos											Revocação %	Precisão %
	Total	Inadequados						Redundantes		Desejáveis			
		Err.	Lin.	Inc.	Mul.	Tot.	%	Qtd.	%	Qtd.	%		
2005/10270	50	2	1			3	6,00	17	34,00	8	16,00	78,95	60,00
2005/4523	31	2				2	6,45			10	32,26	74,36	93,55
2005/4510	40	2			1	3	7,50			5	12,50	88,10	92,50
2005/4565	30	3		2		5	16,67			6	20,00	80,65	83,33
2005/4604	30	1				1	3,33	2	6,67	5	16,67	84,38	90,00
2005/5413	50	3				3	6,00	5	10,00	6	12,00	87,50	84,00
2005/5927	44	3				3	6,82	2	4,55	7	15,91	84,78	88,64
2005/10376	31							5	16,13	4	12,90	86,67	83,87
2005/10412	73	7				7	9,59	24	32,88	10	13,70	80,77	57,53
2005/10515	28	3				3	10,71	6	21,43	6	21,43	76,00	67,86

Tabela 11: Resultado da avaliação com enfoque no usuário

(continuação)

Acórdão	Indivíduos											Revocação	Precisão		
	Total	Inadequados						Redundantes		Desejáveis				%	%
		Err.	Lin.	Inc.	Mul.	Tot.	%	Qtd.	%	Qtd.	%				
2005/10485	42	2				2	4,76	16	38,10	2	4,76	92,31	57,14		
2005/10584	36	3				3	8,33	13	36,11	2	5,56	90,91	55,56		
2005/10641	38	6				6	15,79	10	26,32	7	18,42	75,86	57,89		
2005/10649	38	8				8	21,05	4	10,53	7	18,42	78,79	68,42		
2005/10711	28	10				10	35,71			12	42,86	60,00	64,29		
2005/10714	35	4				4	11,43	10	28,57	8	22,86	72,41	60,00		
2005/10740	33	5				5	15,15	3	9,09	11	33,33	69,44	75,76		
2005/10886	29	7			1	8	27,59	2	6,90	6	20,69	76,00	65,52		
2005/10847	95	7			1	8	8,42	46	48,42	10	10,53	80,39	43,16		
2005/10812	94	6			1	7	7,45	44	46,81	2	2,13	95,56	45,74		
2005/15555	26	1				1	3,85			2	7,69	92,59	96,15		
2005/12231	54	3				3	5,56	19	35,19	7	12,96	82,05	59,26		
2005/11225	43	11				11	25,58	4	9,30	7	16,28	80,00	65,12		
2005/20474	35							5	14,29			100,00	85,71		
2005/1612	50	2				2	4,00					100,00	96,00		
2005/1569	70	3				3	4,29	33	47,14			100,00	48,57		
2005/2563	53	6				6	11,32			6	11,32	88,68	88,68		
2005/2664	28	1				1	3,57	4	14,29	13	46,43	63,89	82,14		
2005/2720	25							2	8,00	1	4,00	95,83	92,00		
2005/2744	18			1		1	5,56			3	16,67	85,00	94,44		
2005/2800	26	1		1		2	7,69			5	19,23	82,76	92,31		
2005/3130	70	3				3	4,29			2	2,86	97,10	95,71		
2005/3191	29									7	24,14	80,56	100,00		
2005/3287	29											100,00	100,00		
2005/3320	34											100,00	100,00		
2005/3407	27	1				1	3,70					100,00	96,30		
2005/3550	37									2	5,41	94,87	100,00		
2005/1489	16									2	12,50	88,89	100,00		
2005/1488	16									1	6,25	94,12	100,00		

Tabela 11: Resultado da avaliação com enfoque no usuário

(continuação)

Acórdão	Indivíduos											Revocação	Precisão		
	Total	Inadequados						Redundantes		Desejáveis				%	%
		Err.	Lin.	Inc.	Mul.	Tot.	%	Qtd.	%	Qtd.	%				
2005/3570	41	3				3	7,32			9	21,95	80,85	92,68		
2005/3711	32	2				2	6,25			6	18,75	83,33	93,75		
2005/4297	40	1			1	2	5,00	1	2,50	6	15,00	86,05	92,50		
2005/528	25	2				2	8,00			4	16,00	85,19	92,00		
2005/159	32	5				5	15,63			3	9,38	90,00	84,38		
2005/673	23	2				2	8,70			6	26,09	77,78	91,30		
2005/797	26	2				2	7,69			1	3,85	96,00	92,31		
2005/835	24	2				2	8,33			7	29,17	75,86	91,67		
2005/1331	35	1				1	2,86			8	22,86	80,95	97,14		
2005/1469	31	2				2	6,45			4	12,90	87,88	93,55		
TOTAL	1870	138	1	4	5	148	7,91	277	14,81	246	13,16	85,45	77,27		

Onde:

Acórdão indica o código do *acórdão* analisado;

Indivíduos expressa em números o resultado da população da ontologia bem como da análise dos *indivíduos* extraídos de cada acórdão;

Total indica a quantidade de indivíduos extraídos do acórdão e populados na ontologia;

Inadequados apresenta a quantidade de indivíduos que foram julgados *inadequados* pelo especialista, conforme a sua análise do texto integral do acórdão. Visando a uma análise mais apurada do resultado, as inadequações foram tipificadas da seguinte maneira:

Err. quantidade de indivíduos que foram considerados inadequados devido a *incorrekções*, tais como: o indivíduo não devia ter sido extraído; o indivíduo foi populado para o conceito errado, etc;

Lin. quantidade de indivíduos que foram considerados inadequados devido a incompatibilidade da *linguagem*, por exemplo: termos de indivíduos consistindo em derivações que o afastam da terminologia padrão existente no tesouro, provocando a subsunção a um conceito inadequado;

Inc. indica que o indivíduo está *incompleto*, ou seja, quando da extração do indivíduo alguns termos importantes foram deixados de fora;

Mul. indica indivíduos *múltiplos*, ou seja, foi considerado que, apesar de correto, a sentença original deveria ter sido dividida em mais de um indivíduo;

Tot. indica a quantidade *total* de indivíduos considerados inadequados;

% indica o *percentual do total* de indivíduos que foram considerados inadequados;

Redundantes demonstra a quantidade e o percentual de indivíduos *redundantes*, ou seja, aqueles que foram aceitos pelos especialistas, porém apontados como repetidos por terem conteúdo igual ou similar a outros indivíduos do mesmo acórdão. Nesta quantidade está descontada a primeira ocorrência;

Desejáveis é a quantidade e o percentual (em relação ao total de conceitos populados) de indivíduos *desejáveis*, segundo a opinião dos especialistas, ou seja, sentenças que constam no texto do acórdão, que não foram populadas em conceito algum, mas que deveriam tê-lo sido. Os quesitos *redundante* e *desejável* são exclusivos, ou seja um indivíduo nunca será redundante e desejável ao mesmo tempo, posto que no primeiro caso se errou pelo excesso – o indivíduo se repetiu no conceito – enquanto que no segundo se errou pela falta – o indivíduo não existiu (contrariando o que se esperava);

Revocação é a taxa de *revocação* da população de conceitos do acórdão, a qual demonstra o percentual de conceitos populados corretamente pelo sistema **dentro do conjunto esperado**. Como para esta avaliação não havia dados de aprendizado prévio, os quais poderiam determinar o que se espera do resultado, foi adotado o seguinte método para determinar o conjunto esperado: o total populado é subtraído dos indevidos e dos redundantes e somado aos desejáveis. Assim, a taxa de revocação pode ser equacionada da seguinte forma:

$$C = \frac{t - i - r}{t - i - r + d} \cdot 100$$

onde:

C é o percentual de revocação resultante;

- t** é o total de indivíduos extraídos pela aplicação;
- i** é o total de indivíduos considerados inadequados;
- r** é a quantidade de indivíduos considerados redundantes;
- d** é a quantidade de indivíduos desejáveis;
- t - i - r** quantidade de indivíduos considerados corretos;
- t - i - r + d** quantidade de indivíduos esperados;

Precisão indica a taxa de *precisão* da população de conceitos do acórdão, a qual demonstra o percentual de conceitos populados corretamente **dentro do conjunto gerado pelo sistema**, sendo definida pela seguinte equação:

$$P = \frac{t - i - r}{t} \cdot 100$$

onde:

- P** é o percentual de precisão resultante;
- t** é o total de indivíduos extraídos pela aplicação;
- i** é o total de indivíduos considerados inadequados;
- r** é a quantidade de indivíduos considerados redundantes;
- t - i - r** quantidade de indivíduos considerados corretos;

4.2 AVALIAÇÃO DA PESQUISA SEMÂNTICA

Foram efetuadas pesquisas semânticas nos três A-Box da ontologia, totalizando quinze mil acórdãos populados para se corresponder à T-Box definida, visando à busca por temas de grande impacto para a atual sociedade brasileira. As pesquisas tiveram uma maior concentração no conceito Legislação, visto que, segundo os especialistas, consiste nas buscas mais frequentes dos usuários de sistemas de jurisprudências. Os temas abordados foram: Código Penal; Estatuto da Criança e do Adolescente (ECA), lei nº 8.069 de 13 de julho de 1990; Lei Maria da Penha, nº 11.340 de 7 de agosto de 2006; Estatuto do Desarmamento, lei nº 10.826, de 22 dezembro de 2003.

Nas pesquisas foram utilizadas duas abordagens: na primeira realizaram-se buscas pelas legislações isoladas (através do conector lógico OU) e na segunda, buscas visando ao cruzamento das leis – aqueles que faziam sentido – (através do conector lógico E). As pesquisas, bem como a análise dos seus resultados, foram processadas conforme descrito na seção 3.4,

página 137, culminando nos dados tabulados conforme a Tabela 12.

Tabela 12: Resultado da avaliação da pesquisa

Legislação	Expressões de busca	Rela	Onto	Cons	Rev	Prec
Código Penal	OU:Legislacao:Nome:código penal	1.886	1.870	1.870	99,15	100,00
ECA (8.069)	OU:Legislacao:Numero:8069	87	76	75	86,21	98,68
M ^a da Penha (11.340)	OU:Legislacao:Nome:penha OU:Legislacao:Numero:11340	39	36	36	92,31	100,00
Desarmamento (10.826)	OU:Legislacao:Nome:desarmamento OU:Legislacao:Numero:10826	377	367	366	97,08	99,73
Sub-total		2.389	2.349	2.347	98,24	99,91
Média parcial					93,69	99,60
ECA e Código Penal	E:Legislacao:Numero:8069 E:Legislacao:Nome:código penal	42	35	35	83,33	100,00
M ^a da Penha e Código Penal	E:Legislacao:Numero:11340 E:Legislacao:Nome:código penal	33	32	32	96,97	100,00
Desarmamento e Código Penal	E:Legislacao:Numero:10826 E:Legislacao:Nome:código penal	217	210	209	96,31	99,52
M ^a da Penha e ECA	E:Legislacao:Numero:11340 E:Legislacao:Numero:8069	9	9	9	100,00	100,00
Sub-total		301	286	285	94,68	99,65
Média parcial					94,15	99,88
Total		2.690	2.635	2.632	97,84	99,89
Média					93,92	99,74

onde:

Legislação informa qual (ou quais) legislação foi almejada na pesquisa;

Expressões de busca são as expressões que foram montadas pela interface do módulo de pesquisa a partir dos requisitos especificados pelo usuário, conforme descrito na página 130, e que foram, na sequência transformadas em consulta SPARQL sendo submetidas à ontologia;

Rela é a quantidade de tuplas retornadas da execução da correspondente consulta SQL efetuada nas **relações** que contém os corpus. Esta quantidade representa a quantidade de dados relevantes (ou esperados);

Onto é a quantidade de indivíduos retornados da pesquisa à **ontologia**;

Cons é a quantidade de indivíduos retornados pela pesquisa à ontologia que foram considerados **consistentes** ao ter a sua propriedade URL comparada (visualmente pelo autor) com as tuplas retornadas da pesquisa relacional;

Rev é o índice de *revocação* alcançado pela pesquisa à ontologia em relação à pesquisa relacional, calculado conforme o método descrito na página 141 e baseado nos valores de *Rela* e *Cons*;

Prec é o índice de *precisão* alcançado pela pesquisa à ontologia, calculado conforme o método descrito na página 141 e baseado nos valores de *Onto* e *Cons*;

Sub-total compreende o somatório das colunas *Rela*, *Onto* e *Cons*, dentro do sub-grupo, e os valores recalculados de *Rev* e *Prec*, baseados nos somatórios obtidos;

Média parcial compreende a média aritmética dos valores de *Rev* e *Prec*, dentro do sub-grupo;

Total compreende o somatório das colunas *Rela*, *Onto* e *Cons*, em ambos sub-grupos, e os valores recalculados de *Rev* e *Prec*, baseados nos somatórios obtidos;

Média compreende a média aritmética dos valores de *Rev* e *Prec*, envolvendo os dois sub-grupos;

4.3 DISCUSSÃO DOS RESULTADOS

A tabulação dos resultados da análise dos especialistas sobre a extração de indivíduos possibilitou o cálculo do índice de revocação da conceitualização resultante do aplicativo, a qual totalizou 85,45%, e do seu índice de precisão, o qual totalizou 77,27%. A precisão resultou em um percentual inferior à revocação devido a uma variável no seu cálculo introduzida: os indivíduos considerados redundantes pelos especialistas, que não os consideraram incorretos, mas que afetam a precisão do resultado, tendo sido assim descontados.

Dada a inexistência de sistemas de indexação automática no poder judiciário e de sistemas de conceitualização automática dentre os trabalhos correlatos pesquisados, não há dados preliminares disponíveis para comparação com os percentuais aqui obtidos. Entretanto, para um sistema de Inteligência Artificial estes resultados se demonstram muito promissores, principalmente se considerado que o aplicativo não dispunha de um aprendizado de máquina prévio e nem tão pouco requereu intervenção do usuário para chegar aos resultados.

A segunda validação visou a avaliar a qualidade dos resultados das pesquisas semânticas, realizadas através do aplicativo, onde obtiveram-se médias de revocação de 93,92% e de precisão de 99,74%. Como foi visto na seção 3.4, a revocação da pesquisa atual do TJPR é desconhecida e não há disponibilidade de especialistas para determinar manualmente o conjunto esperado para o resultado de ambos os sistemas (o que motivou a determinação do conjunto de

resultados esperados baseado em consultas ao corpus carregado em uma base de dados relacional). Devido a isso não foi possível comparar os índices de precisão e revocação da pesquisa semântica com os da pesquisa atual textual do TJPR.

Entretanto, conforme apresentado no Apêndice A, foram determinadas a revocação e a precisão das pesquisas restritas baseadas em índices gerados manual e previamente, disponíveis apenas em três tribunais: STJ (97,37 e 100), STF (66,67 e 100) e TJDFT (15,68 e 100). Nota-se que na precisão houve um empate técnico entre aquelas três pesquisas e a deste trabalho. Mas o grande diferencial está na revocação, onde chegou-se a obter uma vantagem de 78,24% para a pesquisa deste trabalho, que só teve desvantagem (pequena de 3,45%) em relação à pesquisa do STJ. Deve-se ainda considerar neste comparativo, que a pesquisa semântica aqui produzida não contou com inferências humanas na indexação e, ainda assim, teve uma vantagem de 34,02% em relação à revocação média das pesquisas dos referidos tribunais (59,90%).

É possível também efetuar uma comparação com os índices obtidos em alguns trabalhos correlatos que também visavam ao aprimoramento das pesquisas correntes. Assumpção e Neto (2001) obteve, entre os vários experimentos realizados, a melhor relação Revocação / Precisão nos percentuais 70% e 60%. Oliveira (2008) obteve uma precisão de 54%. Júnior (2007) alcançou uma revocação de 85% e precisão 96,66%. Seria uma conclusão insipiente afirmar que o presente trabalho foi mais bem sucedido baseado no fato que aqui se alcançaram percentuais superiores aos outros trabalhos. Uma comparação correta demandaria de uma metodologia científica para garantir uma mensuração imparcial e equitativa dos resultados de todos os trabalhos (o que não faz parte do escopo deste trabalho), sendo que a simples observação numérica dos outros resultados pode servir no máximo de ponto de referência para se avaliar os resultados presentes, os quais demonstraram que a pesquisa semântica aqui produzida alcançou um alto nível de qualidade.

5 CONCLUSÃO

Almejando a indexação e a conceitualização automática de acórdãos, com o resultado destas operações disponibilizado em uma base de conhecimentos a ser pesquisada semanticamente, neste trabalho foram revisadas, e selecionadas, técnicas de recuperação de textos bem como metodologias para construção e gestão de bases de conhecimento. Em seguida, foram elaboradas propostas de métodos para a construção de uma ontologia de aplicação para representação de acórdãos, bem como de desenvolvimento de um aplicativo para a população e pesquisa da mesma ontologia. Para demonstrar a viabilidade de tais métodos foram construídos a ontologia **JurisTJPR** e o aplicativo **jurTreevis**. O primeiro artefato define uma terminologia suficiente para representar os conceitos comumente encontrados nos acórdãos do TJPR e que se pretende disponibilizar em uma pesquisa. O segundo consiste em um sistema capaz de carregar em memória a ontologia definida, carregar um corpus de acórdãos, processar os documentos do corpus visando à extração de indivíduos para a terminologia, popular os indivíduos extraídos na ontologia e por fim efetuar pesquisas semânticas na ontologia populada.

Para avaliar se os objetivos (citados no parágrafo anterior) foram alcançados, duas validações foram efetuadas. A primeira visou a avaliar a qualidade da conceitualização efetuada sobre os acórdãos, sendo para isso definida uma amostra de cinquenta documentos a qual foi submetida ao módulo do aplicativo de extração e população da ontologia. A conceitualização resultante de cada acórdão foi formatada em um relatório sendo submetido à apreciação de especialistas em jurisprudência. A tabulação dos resultados da análise dos especialistas possibilitou o cálculo do índice de revocação da conceitualização resultante do aplicativo, a qual totalizou 85,45%, e do seu índice de precisão, o qual totalizou 77,27%. A segunda validação visou a avaliar a qualidade dos resultados das pesquisas semânticas realizadas através do aplicativo. Para efetuar também uma análise de revocação e precisão, foi definido o conjunto de dados esperados através de consulta a uma base relacional contendo o mesmo corpus, o qual continha quinze mil acórdãos. Assim, comparando os resultados obtidos em várias pesquisas com os dados esperados para as mesmas, obtiveram-se médias de revocação de 93,92% e de precisão de 99,74%.

Os índices de revocação obtidos são superiores aos da grande maioria das pesquisas basea-

das em indexação manual prévia nos tribunais e maiores do que todos os obtidos nos trabalhos correlatos onde também se validou desempenho de pesquisas. Considerando que o aplicativo não dispunha de um aprendizado de máquina prévio e nem requereu intervenção do usuário para a conceitualização, conclui-se que a abordagem foi satisfatória tanto na indexação dos documentos como na pesquisa semântica, demonstrando que a ontologia desenvolvida responde aos requisitos da aplicação. Assim, é possível afirmar que os objetivos almejados para o presente trabalho foram alcançados.

Por fim, avalia-se que as contribuições científicas almejadas, conforme consta na Seção 1.4, foram realizadas, a saber:

experimentação com massa de dados reais: foram utilizados mais de quinze mil acórdãos da base de dados real do TJPR para compor o corpus dos experimentos;

criação de uma ontologia de aplicação para descrever acórdãos: foi desenvolvida a ontologia JurisTJPR;

conceitualização de documentos jurídicos: foi construída uma aplicação para extração de sentenças de acórdãos com subsequente associação a conceitos do respectivo domínio;

conceitualização automática de documentos: a aplicação não depende de intervenção de especialistas ou usuários para efetuar a conceitualização do corpus de acórdãos;

conceitualização independente de treinamento: a aplicação não se utiliza de treinamento ou de informações a priori para conceitualizar os acórdãos;

entendimento e detalhamento da metodologia OTKM: a explanação do método empregado na construção da ontologia, baseado na metodologia OTKM, pode ser utilizada como guia, ou fonte de consulta de experiências, para outros pesquisadores que venham a se utilizar desta metodologia.

6 SUGESTÕES PARA TRABALHOS FUTUROS

As experimentações efetuadas demonstraram que há possibilidades de aprimoramento dos métodos aqui utilizados. A análise dos especialistas refutou alguns indivíduos e sugeriu outros desejáveis, os quais não foram devidamente processados devido a incompatibilidade da linguagem. Estas não-conformidades sugerem que uma evolução dos resultados deste trabalho seria possível através de um aprofundamento em técnicas linguísticas, visando a obter, por exemplo, um algoritmo de *stemming* mais apurado, bem como por uma análise de similaridade entre indivíduos, por exemplo, baseada em um enriquecimento do tesouro aqui utilizado.

Os especialistas ainda apontaram inadequações de indivíduos devido a redundâncias, ou seja, são aqueles que, apesar de corretamente populados em determinados conceitos, são similares ou idênticos a outros indivíduos já populados para o mesmo conceito. Esta parece ser uma importante lacuna, sem “preenchimento” na literatura utilizada nesta dissertação, cuja solução requer estudos mais aprofundados ou até uma pesquisa de doutorado.

Ainda em relação ao tesouro, foi apontado pelos especialistas que aquele do CJF abrange o direito tão somente da esfera federal, deixando de fora assuntos que afetam exclusivamente os níveis estadual e municipal. Assim, uma ampliação do tesouro para abranger as três esferas do direito propiciaria uma conceitualização mais completa, principalmente se considerado que os acórdãos processados no presente projeto são oriundos da justiça estadual.

Além disso, em trabalhos futuros poderia-se almejar uma evolução na arquitetura do aplicativo desenvolvido, transportando as regras da extração de sentenças, as quais foram codificadas diretamente nos códigos-fonte, para uma ontologia de aplicação adjacente. Isso tornaria o sistema mais encapsulado, logo passível de sofrer evoluções nos padrões de extração de conceitos, sem requerer alterações de código, ficando as alterações restritas à ontologia adjacente. Esta nova arquitetura facilitaria também a adaptação do sistema para outros tribunais que disponham de padrões de conceitos diversos, bastando, novamente, se adaptar apenas a ontologia adjacente.

Outros trabalhos poderiam se utilizar da ontologia aqui desenvolvida para o domínio de acórdãos, visando a adaptá-la ou ampliá-la para o domínio de outros documentos referentes a

decisões judiciais tais como *decisões monocráticas* e *sentenças* o que requereria um esforço em analisar as estruturas, os conceitos e os padrões de sentenças neles existentes, visando a criar novas especializações dos conceitos já existentes na ontologia JurisTJPR.

REFERÊNCIAS

- ALVARES, R. V. **Investigação do Processo de Stemming na Língua Portuguesa**. Dissertação (Mestrado) — Universidade Federal Fluminense, Niterói, 2005.
- ALVES, I. M. D. R. **O uso da semântica verbal em sistemas de extração de informação: a construção de uma ontologia de domínio jurídico**. Dissertação (Mestrado) — Universidade do Vale do Rio dos Sinos, São Leopoldo, 2005.
- ASSUMPÇÃO, R. T.; NETO, B. R. Recuperação de documentos jurídicos baseada em um tesouro. In: **Simpósio Brasileiro de Banco de Dados (SBBD)**. [S.l.: s.n.], 2001. p. 213–227.
- BAADER, F. et al. (Ed.). **The Description Logic Handbook: Theory, Implementation, and Applications**. [S.l.]: Cambridge University Press, 2003. ISBN 0-521-78176-0.
- BEN-ARI, M. **Mathematical Logic for Computer Science, Second Edition**. London: Springer, 2001. ISBN 978-1-85233-319-5.
- BERNERS-LEE, T.; FIELDING, R. T.; MASINTER, L. **Uniform Resource Identifier (URI): Generic Syntax**. The Internet Society, 2005. Disponível em: <<http://labs.apache.org/webarch/uri/rfc/rfc3986.html#intro>>.
- BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The semantic web. **Scientific American**, v. 284, n. 5, p. 34–43, maio 2001. ISSN 0036-8733. Disponível em: <<http://www.scientificamerican.com/article.cfm?id=the-semantic-web>>.
- BERRY, M. W.; CASTELLANOS, M. **Survey of Text Mining II: Clustering, Classification and Retrieval**. Palo Alto: Springer, 2008. 240 p. ISBN 978-1-84800-045-2.
- BRACHMAN, R. J.; LEVESQUE, H. J. **Knowledge Representation and Reasoning**. San Francisco: Morgan Kaufmann Publishers, 2004. 381 p. ISBN 978-1-55860-932-7.
- BRAY, T. et al. **Extensible Markup Language (XML) 1.0**. W3C Recommendation, 2000. Disponível em: <<http://www.w3.org/TR/2000/REC-xml-20001006>>.
- BREUKER, J. et al. **OWL Ontology of Basic Legal Concepts (LKIF-Core)**. [S.l.], 2007. Disponível em: <<http://www.estrellaproject.org/lkif-core>>.
- BRICKLEY, D.; GUHA, R. **RDF Vocabulary Description Language 1.0: RDF Schema**. W3C Recommendation, 2004. Disponível em: <<http://www.w3.org/TR/2004/REC-rdf-schema-20040210/>>.
- BRICKLEY, D.; MILLER, L. **The Friend of a Friend (FOAF) project**. FOAF project, 2000. Disponível em: <<http://www.foaf-project.org/>>.
- BUTTLER, D. et al. Querying multiple bioinformatics information sources: Can semantic web research help. **SIGMOD Record**, v. 31, 2002.

BUZAN, T. **Use your Head**. [S.l.]: BBC Books, 2000. 168 p. ISBN 978-0-56353-729-8.

CEJ. **Tesouro Jurídico da Justiça Federal**. Centro De Estudos Judiciários Do Conselho Da Justiça Federal, 1997. Disponível em: <<http://www.cjf.jus.br/biblioteca/>>.

CERQUEIRA, R. F. P. D. **MÉTODO DE MODELAGEM DOMÍNIO-ONTOLÓGICA DO DIREITO POSITIVO BRASILEIRO**. Dissertação (Mestrado) — Escola de Ciência da Informação da UFMG, Belo Horizonte, 2007.

DIAS, M. A. L. **PortugueseStemmer**. Lajeado/RS: UNIVATES, 2005. Disponível em: <<http://ensino.univates.br/~abadia/portugueseStemmer.java>>.

EDMUNDSON, H. New methods in automatic extraction. **ACM**, v. 16, p. 264–285, 1968.

FALCÃO, S. D. **Proposição, Aplicação e Avaliação de um Método de Classificação Temática em Bases de Dados Textuais Indexadas com auxílio de Vocabulários Controlados**. Dissertação (Mestrado) — Universidade Católica de Brasília, Brasília, 2003.

FARZINDAR, A.; LAPALME, G. Letsum, an automatic legal text summarizing system. **Legal Knowledge and Information Systems. Jurix 2004: The Seventeenth Annual Conference**, p. 11–18, 2004.

FAYYAD, U. M. et al. (Ed.). **Advances in Knowledge Discovery and Data Mining**. [S.l.]: AAAI/MIT Press, 1996. ISBN 0-262-56097-6.

FERNÁNDEZ, M.; GÓMEZ-PÉREZ, A.; JURISTO, N. Methontology: from ontological art towards ontological engineering. In: **Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering**. Stanford, USA: [s.n.], 1997. p. 33–40.

FRANK, E. **StopwordsEnglish**. HAMILTON, NEW ZEALAND: University of Waikato, 2010. Disponível em: <<http://code.google.com/p/zh-powerset/source/browse/trunk/ws/src/LookAtMe/StopwordsEnglish.java?spec=svn67&r=57>>.

FRANK, E. et al. The weka data mining software: An update. **SIGKDD Explorations**, v. 11, p. 10–18, 2009. Disponível em: <<http://www.cs.waikato.ac.nz/ml/weka/>>.

FRIEDL, J. E. F. **Mastering Regular Expressions, Third Edition**. USA: O'Reilly Media, Inc., 2006. 544 p. ISBN 978-0-596-52812-6.

GÓMEZ-PÉREZ, A. Ontological engineering: A state of the art. **Expert Update**, v. 2, n. 3, p. 33–44, 1999.

GUARINO, N. **Formal Ontology in Information Systems: Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy**. 1st. ed. Amsterdam, The Netherlands: IOS Press, 1998. ISBN 9051993994.

GUARINO, N.; WELTY, C. Evaluating ontological decisions with ontoclean. **Communications of the ACM**, v. 45, p. 61–65, 2002.

HEFLIN, J. **OWL Web Ontology Language Use Cases and Requirements**. W3C Recommendation, 2004. Disponível em: <<http://www.w3.org/TR/webont-req/>>.

HEWLETT-PACKARD. **Jena - A Semantic Web Framework for Java**. Source Forge, 2010. Disponível em: <<http://jena.sourceforge.net/>>.

INFORMATICS, S. R. C. F. B. **Protégé**. Stanford University School of Medicine, 2010. Disponível em: <<http://protege.stanford.edu/download/download.html>>.

JARGAS, A. M. **Expressões Regulares – Uma abordagem divertida**. Brasil: Novatec, 2009. 208 p. ISBN 978-85-7522-212-6.

JIZBA, R. **Measuring Search Effectiveness**. Creighton University Health Sciences Library and Learning Resources Center, 2000. Disponível em: <http://www.creighton.edu/fileadmin/user/HSL/docs/ref/Searching_-_Recall_Precision.pdf>.

JÚNIOR, A. T. D. C. **Indexação automática de acórdãos por meio de processamento de linguagem natural**. Dissertação (Mestrado) — Universidade de Brasília, Brasília, 2007.

JÚNIOR, H. S. R.; ROVER, A. J. Ontologias aplicadas ao direito e ao governo eletrônico: experiências e projetos. **Congresso Alagoas Digital**, Inovações Acadêmicas e Projetos de Software Livre, p. 100–108, 2009.

JÚNIOR, M. D. S. B. **Proposta de modelo rbc para a recuperação inteligente de jurisprudência na justiça federal**. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Florianópolis, 2001.

KLYNE, G.; CARROLL, J. J. **Resource Description Framework (RDF): Concepts and Abstract Syntax**. W3C Recommendation, 2004. Disponível em: <<http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>>.

MANOLA, F.; MILLER, E. **RDF Primer**. W3C Recommendation, 2004. Disponível em: <<http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>>.

MAÇOLI, F. **ANÁLISE DOCUMENTÁRIA DAS EMENTAS CÍVEIS - Uma experiência com acórdãos do Tribunal de Justiça de São Paulo**. Dissertação (Mestrado) — Pontifícia Universidade Católica de Campinas, Campinas, 2005.

MCCARTHY, P. **Search RDF data with SPARQL**. IBM Developer Works, 2005. Disponível em: <<http://www.ibm.com/developerworks/xml/library/j-sparql/>>.

MCGUINNESS, D. L.; HARMELEN, F. V. **OWL Web Ontology Language Overview**. W3C Recommendation, 2004. Disponível em: <<http://www.w3.org/TR/2004/REC-owl-features-20040210/#s1.2>>.

MOTIK, B. et al. **OWL 2 Web Ontology Language: Direct Semantics**. 2009. 1–17 p. World Wide Web Consortium, Working Draft WD-owl2-semantics-20081202. Disponível em: <<http://www.w3.org/TR/2008/WD-owl2-semantics-20081202/>>.

MOTTA, D. F. D. **Método Relacional como Nova Abordagem para a Construção de Tesouros**. Dissertação (Mestrado) — Instituto Brasileiro de Informação em Ciência e Tecnologia, Rio de Janeiro, 1986. Disponível em: <<http://www.conexaorio.com/bitl/dilza/index.htm>>.

MYLOPOULOS, J. Information modeling in the time of the revolution. **Information Systems**, Elsevier Science Ltd., Oxford, UK, v. 23, n. 3-4, p. 127–155, 1998. ISSN 0306-4379. Disponível em: <[http://dx.doi.org/10.1016/S0306-4379\(98\)00005-2](http://dx.doi.org/10.1016/S0306-4379(98)00005-2)>.

NETWORK, S. D. **The Java Tutorials – Regular Expressions**. 2010. Disponível em: <<http://java.sun.com/docs/books/tutorial/essential/regex/index.html>>.

NETWORK, S. D. **Java 2 SDK, Standard Edition Documentation**. 2011. Disponível em: <<http://download.oracle.com/javase/1.4.2/docs/>>.

NOY, N. F.; MCGUINNESS, D. L. **Ontology Development 101: A Guide to Creating Your First Ontology**. [S.l.], 2001. Disponível em: <http://www.ksl.stanford.edu/KSL_Abstracts/KSL-01-05.html>.

OLIVEIRA, S. R. D. **RECUPERAÇÃO INTELIGENTE DE JURISPRUDÊNCIA**. Dissertação (Mestrado) — Universidade de Brasília, Brasília, 2008.

PARK, Y.; BYRD, R. J.; BOGURAEV, B. K. Towards ontologies on demand. **IBM T.J. Watson Research Center**, 2003.

PARSIA, L. C. . **Pellet**. 2010. Disponível em: <<http://clarkparsia.com/pellet/download>>.

PRUD'HOMMEAUX, E.; SEABORNE, A. **SPARQL Query Language for RDF**. W3C Recommendation, 2008. Disponível em: <<http://www.w3.org/TR/rdf-sparql-query/>>.

RUSSELL, S. J.; NORVIG, P. **Artificial Intelligence – A Modern Approach**. 3rd. ed. New Jersey: Prentice Hall, 2009. 1152 p. ISBN 978-0-13604-259-4.

SANTOS, W. D. **Dicionário Jurídico Brasileiro**. Belo Horizonte: LIVRARIA DEL REY EDITORA LTDA., 2001. 340 p. ISBN 85.7308-458-8.

SAVOY, J. **IR Multilingual Resources at UniNE**. Université de Neuchâtel, 2010. Disponível em: <<http://members.unine.ch/jacques.savoy/clef/portugueseST2.txt>>.

SEBASTIANI, F. Machine learning in automated text categorization. **ACM Comput. Surv.**, ACM, New York, NY, USA, v. 34, p. 1–47, March 2002. ISSN 0360-0300. Disponível em: <<http://doi.acm.org/10.1145/505282.505283>>.

SHETH, A.; ARPINAR, I. B.; KASHYAP, V. Relationships at the heart of semantic web: Modeling and discovering and validating and exploiting complex semantic relationship. In: **Proc. 29th Annual Conference on Current Trends in Theory and Practice of Informatics (SOFSEM 2002) and Milovyand Czech Replublic**. [S.l.: s.n.], 2002.

SILVA, D. L. D.; SOUZA, R. R.; ALMEIDA, M. B. Comparação de metodologias para construção de ontologias e vocabulários controlados. **SEMINÁRIO DE PESQUISA EM ONTOLOGIA NO BRASIL**, Universidade Federal Fluminense, Niterói, 2008. Disponível em: <<http://www.uff.br/ontologia/index.php>>.

SILVA, D. P. E.; FILHO, N. S.; CARVALHO, G. **Vocabulário Jurídico Conciso**. 2. ed. Rio de Janeiro: Forense Jurídica (Grupo GEN), 2009. 773 p. ISBN 978-85-309-2775-2.

TSE. **Glossário Eleitoral Brasileiro**. Tribunal Superior Eleitoral, 2010. Disponível em: <<http://www.tse.gov.br/internet/institucional/glossario-eleitoral/>>.

YORK, S.; STEFFEN, S.; STUDER, R. On-to-knowledge methodology (otkm). In: **Handbook on Ontologies, International Handbooks on Information Systems**. [S.l.]: Springer, 2003. p. 117–132.

ZAHRA, F. M. **PORONTO - FERRAMENTA PARA CONSTRUÇÃO SEMIAUTOMÁTICA DE ONTOLOGIAS EM PORTUGUÊS**. Dissertação (Mestrado) — Pontifícia Universidade Católica do Paraná, Curitiba, 2009.

APÊNDICE A – COMPARATIVO ENTRE AS PESQUISAS JURISPRUDENCIAIS

A comparação entre as pesquisas jurisprudenciais, verificadas durante o presente trabalho, se deu através de duas abordagens. Na primeira constatou-se as funcionalidades presentes nas mesmas, conforme o Quadro 13.

Sigla do Tribunal	Pesquisa por						
	Texto	Legis.	Index.	Metadados			
				Tipo Proc.	Órgão	Origem	Ator*
STF	X	X	X		X		R
STJ	X	X	X		X		R
TJAC	X			X	X		R
TJAP	X			X		X	R
TJAL**							
TJAM	X			X	X		R
TJBA	X			X	X	X	R
TJCE	X				X		R
TJDFT	X	X	X		X		R
TJES	X				X		R
TJGO	X					X	R
TJMA	X				X		PRV
TJMT	X			X	X		R
TJMS	X			X	X		R
TJMG	X		X				R
TJPA	X				X		R
TJPB**							

(continua)

Quadro 13: Funcionalidades de pesquisa jurisprudencial nos tribunais brasileiros

Sigla do Tribunal	Pesquisa por						
	Texto	Legis.	Index.	Metadados			
				Tipo Proc.	Órgão	Origem	Ator*
TJPR	X				X		R
TJPE	X			X	X		R
TJPI	X			X	X		R
TJRJ	X				X		R
TJRN	X			X	X		R
TJRS	X			X	X	X	R
TJRO	X					X	PRV
TJRR	X						R
TJSC	X			X	X	X	R
TJSP	X			X	X	X	R
TJSE	X			X	X		R
TJTO	X						PR
* P = Parte R = Relator V = Revisor							
** A pesquisa não funcionou							

Quadro 13: Recursos de pesquisa jurisprudencial nos tribunais brasileiros

onde:

Texto indica, se preenchido com X, que está disponível a funcionalidade de pesquisa livre por *texto*;

Legis. indica que está disponível a funcionalidade de pesquisa por *legislação*, a qual deve ser selecionada em uma lista de leis, pré-definida no sítio – a legislação selecionada será buscada nos documentos que tiverem sido associados previamente à lei, tarefa esta de responsabilidade de especialistas em jurisprudência;

Index. indica que está disponível a funcionalidade de pesquisa por *indexação*, para a qual devem ser informadas palavras a serem buscadas, as quais serão procuradas entre as palavras-chave previamente associadas a cada documento, tarefa esta também de responsabilidade dos especialistas;

Metadados indica que está disponível a funcionalidade de pesquisa por *metadados*, os quais devem ter sido informados para cada documento, quando de sua criação. Os principais tipos de metadados que costumam estar disponíveis para restringir o retorno da busca são:

Tipo Proc., através do qual se pode selecionar, em uma lista, qual o *tipo de processo* dos recursos cujos documentos de jurisprudência se procuram;

Órgão, através do qual se pode selecionar, em uma lista, qual o *órgão julgador* que emitiu o documento;

Origem, através do qual se pode selecionar, em uma lista, a *origem*, ou seja, a comarca de onde vieram as sentenças, as quais foram julgadas através dos documentos jurisprudenciais que se procuram;

Ator, para o qual se pode informar um *ator* do recurso. Geralmente trata-se do Relator do recurso, mas em algumas pesquisas está disponível a busca também por Parte do processo (reclamante ou reclamada) e Revisor do documento.

A segunda abordagem consistiu em uma pesquisa igual em todos os sítios, buscando acórdãos que se refiram à lei nº 11.340/06, batizada de Lei Maria da Penha, a qual se tornou muito popular desde a sua promulgação e que vem gerando inúmeros processos criminais posto que destina-se a coibir um crime muito recorrente – a violência doméstica contra a mulher.

Nas pesquisas livres por texto, utilizaram-se as palavras-chave MARIA e PENHA. Nos sítios onde havia a funcionalidade de pesquisa específica por legislação, buscou-se referência à lei em questão, visando à um resultado mais refinado. Por fim, onde estava disponível a funcionalidade de pesquisa por indexação, utilizaram-se as mesmas palavras da pesquisa livre.

O estudo se iniciou pelos sítios cujas buscas dispõem de mais funcionalidades: STF, STJ, TJDFT e TJMG – como se pode ver no Quadro 13. Nestas pesquisas foram colhidos os seguintes resultados:

- no STJ a pesquisa livre resultou em quarenta e seis documentos, nos quais os termos aparecem como Legislação em trinta e oito documentos, como Foro¹ em seis documentos e como Nome de Parte em dois documentos. Já a pesquisa pela legislação “LEI MARIA DA PENHA (LMP-06)” resultou em dezenove documentos em cuja Legislação consta corretamente a lei desejada. Finalmente, a pesquisa por indexação resultou em trinta e sete documentos, todos correlatos à legislação desejada;

¹Foro é a localidade de julgamento em primeira instância, geralmente designado pelo nome da cidade sede da comarca.

- no STF a pesquisa livre resultou em trinta documentos, nos quais os termos aparecem somente em nomes de partes dos processos. A pesquisa por Legislação não foi possível pois não consta da lista de legislações a selecionar. Finalmente, a pesquisa por indexação resultou em zero documentos. Nota-se aí, que a legislação pesquisada não se enquadra nas matérias julgadas por aquele tribunal. Assim, para validar a ferramenta de pesquisa, mudou-se a legislação pesquisada para o peculiar Decreto Imperial. A pesquisa livre pelas palavras DECRETO e IMPERIAL resultou em oito documentos, dos quais apenas três são coerentes com a legislação procurada. Na pesquisa pela legislação “DECRETO IMPERIAL”, o sistema pesquisou por um código pré-definido, “DEI-”, resultando em quinze documentos em cuja legislação nada consta sobre aquela desejada e nos quais, por outro lado, a sigla DEI aparece com significados diversos. Finalmente, a pesquisa por indexação resultou em dois documentos, ambos correlatos à legislação desejada;
- no TJDFT a pesquisa livre resultou em trezentos e setenta documentos, todos correlatos à legislação procurada. A pesquisa específica por Legislação não foi possível por não conter na lista de seleção a lei desejada. Já a pesquisa por indexação resultou em cinquenta e oito documentos, todos referentes à legislação desejada;
- no TJMG a pesquisa livre resultou em dois mil quatrocentos e noventa e dois documentos correlatos a informações diversas, além da legislação desejada. Já a pesquisa indexada retornou cento e trinta e seis documentos, todos coerentes com a legislação desejada.

Tabulando os resultados das quatro pesquisas obtém-se a Tabela 14, onde:

Tabela 14: Pesquisas nos sítios com mais funcionalidades

Sigla Tribu.	Pesquisa Livre			Pesquisa por Legislação				Pesquisa por Indexação			
	TL	QLL	%PL	TG	QGL	%PG	%CG	TI	QIL	%PI	%CI
STJ	46	38	82,61	19	19	100,00	50,00	37	37	100,00	97,37
STF	8	3	37,50	15	0	0,00	0,00	2	2	100,00	66,67
TJDFT	370	370	100,00	0	0	0,00	0,00	58	58	100,00	15,68
TJMG	2492							136	136	100,00	
MÉDIA			73,37			33,33	16,67			100,00	59,90

TL indica o *total* de documentos retornados pela pesquisa *livre*;

QLL indica a *quantidade* de documentos, dentre o total retornado pela pesquisa *livre*, que contém referência à *lei* pesquisada;

%PL indica a *precisão* da pesquisa *livre*, ou seja, o percentual, do total retornado, de documentos que contém referência à lei pesquisada, sendo que o seu cálculo pode ser representado através da seguinte equação:

$$\%PL = \frac{QLL}{TL} \cdot 100$$

TG indica o *total* de documentos retornados pela pesquisa por *legislação*;

QGL indica a *quantidade* de documentos, dentre o total retornado pela pesquisa por *legislação*, que contém referência à *lei* pesquisada;

%PG indica a *precisão* da pesquisa por *legislação*, ou seja, o percentual, **do total retornado**, de documentos que contém referência à lei pesquisada, sendo que o seu cálculo pode ser representado através da seguinte equação:

$$\%PG = \frac{QGL}{TG} \cdot 100$$

%CG indica a *revocação* da pesquisa por *legislação*, ou seja, o percentual, **da quantidade esperada**, de documentos que contém referência à lei pesquisada, sendo que o seu cálculo se baseia nas seguintes premissas:

- os documentos retornados pela pesquisa livre contendo referências à legislação pesquisada (quantificado acima por QLL) consistem no conjunto de ocorrências esperadas (supondo que o resultado da busca contempla de fato todas as ocorrências da legislação na base de dados);
- os documentos retornados pela pesquisa por legislação contendo efetivas referências à legislação pesquisada (quantificada acima por QGL) consistem no conjunto de indivíduos corretos retornados;
- a definição dos indivíduos corretos se baseou no julgamento do autor, segundo a sua experiência.

Assim o cálculo da revocação pode ser representado através da seguinte equação:

$$\%CG = \frac{QGL}{QLL} \cdot 100$$

TI indica o *total* de documentos retornados pela pesquisa por *indexação*;

QIL indica a *quantidade* de documentos, dentre o total retornado pela pesquisa por *indexação*, que contém referência à *lei* pesquisada;

%PI indica a *precisão* da pesquisa por *indexação*, ou seja, o percentual, do total retornado, de documentos que contém referência à lei pesquisada, sendo que o seu cálculo pode ser representado através da seguinte equação:

$$\%PI = \frac{QIL}{TI} \cdot 100$$

%CI indica a *revocação* da pesquisa por *indexação*, ou seja, o percentual, da quantidade esperada, de documentos que contém referência à lei pesquisada, sendo que o seu cálculo se baseia nas mesmas premissas propostas para o cálculo de %CG, sendo representado pela seguinte equação:

$$\%CI = \frac{QIL}{QLL} \cdot 100$$

O resultado da pesquisa livre no TJMG, por ter sido muito extenso, inviabilizou a análise de quantidade de documentos que se referenciavam à lei pesquisada (QLL), impedindo o cálculo da precisão desta pesquisa (%PL). A pesquisa por legislação era inexistente. Já a pesquisa por indexação foi possível de se executar e avaliar a precisão (%QIL), entretanto o cálculo da sua revocação (%CI) não foi possível devido à falta do QLL.

Demonstrando graficamente os percentuais de precisão e de revocação apresentados na Tabela 14, obtém-se a Figura 27, onde nota-se que a pesquisa livre tem sempre uma vantagem quantitativa, ou seja retorna mais documentos desejados do que as demais pesquisas. Entretanto misturados vêm vários documentos indesejados. Esta desvantagem pode ser percebida se comparadas as colunas (de cores azul, vermelho e roxo) de precisão das quatro consultas aos tribunais, demonstrando que os resultados das pesquisas por legislação e indexação são, geralmente, mais precisas. Por fim, as colunas (de cores verde e ciano) de revocação confirmam o senso comum de que qualidade e quantidade são métricas distintas, pois, apesar de geralmente as pesquisas por legislação e indexação retornarem exclusivamente a legislação desejada, por outro lado elas têm perdas se comparadas à pesquisa livre, retornando menos do que a quantidade de documentos esperada.

O gráfico da Figura 27 também demonstra que a pesquisa por legislação, a qual teve uma revocação média de 16,67%, ao contrário do que se esperava de uma pesquisa avançada, teve um desempenho sofrível. Já a pesquisa por indexação teve altos e baixos, oscilando de uma ótima revocação de 97,37% até uma revocação inaceitável de 15,68%, apresentando, por fim, uma revocação média razoável de 59,90%. Assim, ambas as pesquisas demonstraram que, nem sempre, a especialização de uma ferramenta promove aumento na qualidade do seu resultado.

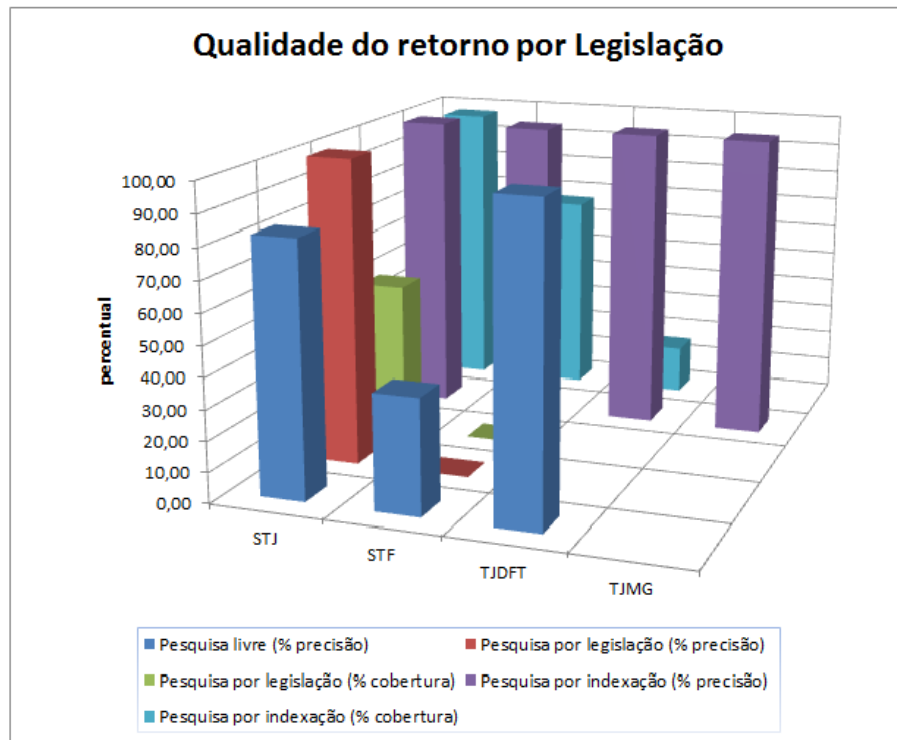


Figura 27: Resultados das pesquisas nos sítios com mais funcionalidades

Nos demais tribunais não houve como efetuar um comparativo, seguindo o método utilizado para os quatro tribunais anteriores, visto que só há disponibilidade de um tipo de pesquisa, a livre. Entretanto, nestes também foi possível verificar a dificuldade de se obter os resultados desejados dada a sua extensão e heterogeneidade, como pode-se ver na tabulação das pesquisas realizadas em alguns dos tribunais, conforme o Quadro 15.

Tribunal	Total	Legislação	Partes	Julgador	Localidades	Texto
TJAC	153	X	X	X		
TJAM	0					
TJAP	11	X				
TJBA	90	X	X		X	
TJCE	20	X	X			
TJES	218	X	X			
TJGO	89	X	X			X
TJMA	17	X				X
TJPR	73	X	X			

Quadro 15: Resultado da pesquisa em outros tribunais

Onde:

Total indica o total de documentos retornados pela pesquisa;

Legislação indica se, dentre os retornados, há documentos com referências à lei pesquisada;

Partes indica se, dentre os retornados, há documentos incluídos devido a referências a partes do processo cujo nome contém os termos pesquisados;

Julgador indica se, dentre os retornados, há documentos incluídos devido a referências a julgadores do acórdão cujo nome contém os termos pesquisados;

Localidades indica se, dentre os retornados, há documentos incluídos devido a referências a localidades cujo nome contém os termos pesquisados;

Texto indica se, dentre os retornados, há documentos incluídos devido a referências a palavras que contém os termos pesquisados, por exemplo **desempenham**, **primaria**, etc.

**APÊNDICE B – RESULTADO DO MÓDULO DE EXTRAÇÃO – EXEMPLO
RESULTANTE DO PROCESSAMENTO DE UM ACÓRDÃO**

```

< class:Acordao >
  < dataType:URL >
    C:\ONTO\corpus\acordao9999999.txt
  < /dataType:URL >
  < object:temApelacao inverse=ehApelacaoDe >
    < class:Apelacao >
      < object:temComarca inverse=ehComarcaDe >
        < class:Comarca >
          < dataType:Nome >
            apucarana
          < /dataType:Nome >
        < /class:Comarca >
      < /object:temComarca >
      < object:temCompetencia >
        < class:Civel >
        < /class:Civel >
      < /object:temCompetencia >
      < dataType:Numero > 999.999-9 < /dataType:Numero >
      < object:temEscrivania inverse=ehEscrivaniaDe >
        < class:Escrivania >
          < dataType:Nome > primeira vara cível < /dataType:Nome >
        < /class:Escrivania >
      < /object:temEscrivania >
    < /class:Apelacao >
  < /object:temApelacao >

```

```

< object:temAtor inverse=ehAtorDe >
  < class:Ator >
    < dataType:Nome >
      banco xxxxxxxxx xxxxxxxxxxx s/a
    < /dataType:Nome >
    < object:temApelante >
      < class:Apelante >
      < /class:Apelante >
    < /object:temApelante >
  < /class:Ator >
< /object:temAtor >
< object:temAtor inverse=ehAtorDe >
  < class:Ator >
    < dataType:Nome >
      município apucarana
    < /dataType:Nome >
    < object:temApelado >
      < class:Apelado >
      < /class:Apelado >
    < /object:temApelado >
  < /class:Ator >
< /object:temAtor >
< object:temAtor inverse=ehAtorDe >
  < class:Ator >
    < dataType:Nome >
      des.a vilma régia ramos rezende
    < /dataType:Nome >
    < object:temRelator >
      < class:Relator >
      < /class:Relator >
    < /object:temRelator >
  < /class:Ator >
< /object:temAtor >

```

```

< object:temFato inverse=ehFatoDe >
  < class:Fato >
    < dataType:Descricao >
      CONSTITUCIONAL E TRIBUTÁRIO
    < /dataType:Descricao >
  < /class:Fato >
< /object:temFato >
< object:temAssunto inverse=ehAssuntoDe >
  < class:Assunto >
    < dataType:Descricao >
      EMBARGOS À EXECUÇÃO
    < /dataType:Descricao >
  < /class:Assunto >
< /object:temAssunto >
< object:temAssunto inverse=ehAssuntoDe >
  < class:Assunto >
    < dataType:Descricao >
      EXECUÇÃO FISCAL
    < /dataType:Descricao >
  < /class:Assunto >
< /object:temAssunto >
< object:temAssunto inverse=ehAssuntoDe >
  < class:Assunto >
    < dataType:Descricao >
      IMPOSTO SOBRE SERVIÇOS DE QUALQUER NATUREZA
    < /dataType:Descricao >
  < /class:Assunto >
< /object:temAssunto >
< object:temAssunto inverse=ehAssuntoDe >
  < class:Assunto >
    < dataType:Descricao >
      CERTIDÃO DA DÍVIDA ATIVA
    < /dataType:Descricao >
  < /class:Assunto >
< /object:temAssunto >

```

```

< object:temAssunto inverse=ehAssuntoDe >
  < class:Assunto >
    < dataType:Descricao >
      ATO ADMINISTRATIVO
    < /dataType:Descricao >
  < /class:Assunto >
< /object:temAssunto >
< object:temAssunto inverse=ehAssuntoDe >
  < class:Assunto >
    < dataType:Descricao >
      PRESUNÇÃO DE LIQUIDEZ E CERTEZA
    < /dataType:Descricao >
  < /class:Assunto >
< /object:temAssunto >
< object:temFato inverse=ehFatoDe >
  < class:Fato >
    < dataType:Descricao >
      SERVIÇOS BANCÁRIOS
    < /dataType:Descricao >
  < /class:Fato >
< /object:temFato >
< object:temFato inverse=ehFatoDe >
  < class:Fato >
    < dataType:Descricao >
      INTERPRETAÇÃO AMPLA E ANALÓGICA
    < /dataType:Descricao >
  < /class:Fato >
< /object:temFato >
< object:temFato inverse=ehFatoDe >
  < class:Fato >
    < dataType:Descricao >
      INCIDÊNCIA NAS CONTAS “TAXA DE MANUTENÇÃO
      DE CONTA CORRENTE”, “TAXA DE MANUTENÇÃO DE POU-
      PANÇA”, “TAXA DE MANUTENÇÃO DE CONTA CORRENTE
      ATIVA”, “OUTROS SERVIÇOS” E “OPERAÇÕES ATIVAS”

```

```

        < /dataType:Descricao >
    < /class:Fato >
< /object:temFato >
< object:temVoto inverse=ehVotoDe >
    < class:Voto >
        < dataType:Descricao >
            RECURSO PARCIALMENTE PROVIDO.
        < /dataType:Descricao >
    < /class:Voto >
< /object:temVoto >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:Data >
            406/68
        < /dataType:Data >
        < dataType:Nome >
            dl
        < /dataType:Nome >
    < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:Numero >
            56/87
        < /dataType:Numero >
        < dataType:Nome >
            lc
        < /dataType:Nome >
    < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:Data >
            100/99
        < /dataType:Data >

```

```

        < dataType:Nome >
            lc
        < /dataType:Nome >
    < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:NumItem >
            iii
        < /dataType:NumItem >
        < dataType:TipoItem >
            inciso
        < /dataType:TipoItem >
        < dataType:NumParagrafo >
            5
        < /dataType:NumParagrafo >
        < dataType:NumArtigo >
            2
        < /dataType:NumArtigo >
        < dataType:Numero >
            6830/80
        < /dataType:Numero >
        < dataType:Nome >
            lei
        < /dataType:Nome >
    < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:NumItem >
            iii
        < /dataType:NumItem >
        < dataType:TipoItem >
            inciso
        < /dataType:TipoItem >

```



```
< dataType:NumArtigo >
    202
< /dataType:NumArtigo >
< dataType:Nome >
    código tributário nacional
< /dataType:Nome >
< /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:NumArtigo >
            203
        < /dataType:NumArtigo >
        < dataType:Nome >
            código tributário nacional
        < /dataType:Nome >
    < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:NumArtigo >
            202
        < /dataType:NumArtigo >
        < dataType:Nome >
            código tributário nacional
        < /dataType:Nome >
    < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
    < class:Legislacao >
        < dataType:NumParagrafo >
            5
        < /dataType:NumParagrafo >
```

```

    < dataType:NumArtigo >
      2
    < /dataType:NumArtigo >
    < dataType:Nome >
      lei execuções fiscais
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumArtigo >
      202
    < /dataType:NumArtigo >
    < dataType:Nome >
      ctn
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumParagrafo >
      5
    < /dataType:NumParagrafo >
    < dataType:NumArtigo >
      2
    < /dataType:NumArtigo >
    < dataType:Data >
      6830/80
    < /dataType:Data >
    < dataType:Nome >
      lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >

```

```

< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumArtigo >
      202
    < /dataType:NumArtigo >
    < dataType:Nome >
      código tributário nacional
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumParagrafo >
      5
    < /dataType:NumParagrafo >
    < dataType:NumArtigo >
      2
    < /dataType:NumArtigo >
    < dataType:Numero >
      6830/80
    < /dataType:Numero >
    < dataType:Nome >
      lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Numero >
      9250/95
    < /dataType:Numero >
    < dataType:Nome >
      lei federal
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >

```

```

< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Numero >
      11580/96
    < /dataType:Numero >
    < dataType:Nome >
      lei estadual
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Numero >
      1025/69
    < /dataType:Numero >
    < dataType:Nome >
      lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumArtigo >
      1
    < /dataType:NumArtigo >
    < dataType:Numero >
      1025/699
    < /dataType:Numero >
    < dataType:Nome >
      lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >

```

```

< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Data >
      406/68
    < /dataType:Data >
    < dataType:Nome >
      decreto-lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Data >
      406/68
    < /dataType:Data >
    < dataType:Nome >
      decreto-lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumItem >
      i
    < /dataType:NumItem >
    < dataType:TipoItem >
      anexo
    < /dataType:TipoItem >
    < dataType:NumItem >
      95
    < /dataType:NumItem >
    < dataType:TipoItem >
      item
    < /dataType:TipoItem >

```

```

    < dataType:Data >
      406/68
    < /dataType:Data >
    < dataType:Data >
      15/12/1968
    < /dataType:Data >
    < dataType:Nome >
      decreto-lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Numero >
      56/87
    < /dataType:Numero >
    < dataType:Nome >
      lc
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Numero >
      1968
    < /dataType:Numero >
    < dataType:Nome >
      decreto
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >

```

```

< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:NumItem >
      i
    < /dataType:NumItem >
    < dataType:TipoItem >
      anexo
    < /dataType:TipoItem >
    < dataType:NumItem >
      95
    < /dataType:NumItem >
    < dataType:TipoItem >
      item
    < /dataType:TipoItem >
    < dataType:Data >
      406/68
    < /dataType:Data >
    < dataType:Nome >
      decreto-lei
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< object:temLegislacao inverse=ehLegislacaoDe >
  < class:Legislacao >
    < dataType:Numero >
      56/87
    < /dataType:Numero >
    < dataType:Nome >
      lc
    < /dataType:Nome >
  < /class:Legislacao >
< /object:temLegislacao >
< /class:Acordao >

```


**APÊNDICE C – RESULTADO DO MÓDULO DE POPULAÇÃO – EXEMPLO
RESULTANTE DO PROCESSAMENTO DE UM ACÓRDÃO**

Marque um X nas lacunas correspondentes às extrações com as quais você não concorda.

- [] Acordao - URL: C:\TEMP\teste\acordao2005\Acordao_2005_99999.RTF
- [] Apelacao - Criminal
- [] - Numero: 999999-9
- [] - Escrivania: 6ª Vara criminal
- [] Ator - Nome: ministério público
- [] - Apelante
- [] Ator - Nome: xxxxxxxxxxx xxxxxx xxxx¹
- [] - Apelado
- [] Ator - Nome: xxxxxxxx xxxxxxxxxxx xxxxx
- [] - Relator
- [] Assunto - Descricao: APELAÇÃO CRIMINAL
- [] Assunto - Descricao: ATENTADO VIOLENTO AO PUDOR
- [] Assunto - Descricao: SENTENÇA ABSOLUTÓRIA
- [] Fato - Descricao: RECURSO
- [] Assunto - Descricao: NOVA DEFINIÇÃO JURÍDICA
- [] Fato - Descricao: IMPOSSIBILIDADE
- [] Fato - Descricao: CONHECIMENTO DO MÉRITO EM SENTIDO AMPLO
- [] Fato - Descricao: AUTORIA DEMONSTRADA
- [] Fato - Descricao: TESTEMUNHO DA VÍTIMA
- [] Fato - Descricao: VALIDADE
- [] Fato - Descricao: MATERIALIDADE NÃO CARACTERIZADA
- [] Fato - Descricao: TENTATIVA INOCORRÊNCIA
- [] Assunto - Descricao: DESISTÊNCIA VOLUNTÁRIA
- [] Fato - Descricao: RECONHECIMENTO

¹Os nomes foram mascarados para evitar quebra de sigilo de justiça.

- [] Voto - Descricao: DECISÃO RECORRIDA MANTIDA.
- [] Legislacao - NumItem: ii
 - [] - TipoItem: inciso
 - [] - NumArtigo: 14
 - [] - NumArtigo: 225
 - [] - Nome: código penal
- [] Legislacao - NumItem: ii
 - [] - TipoItem: inciso
 - [] - Nome: código penal
- [] Legislacao - NumItem: iii
 - [] - TipoItem: incisos
 - [] - NumArtigo: 386
 - [] - Nome: código processo penal
- [] Legislacao - NumArtigo: 214
 - [] - Nome: código penal
- [] Legislacao - NumArtigo: 9
 - [] - Numero: 8072/90
 - [] - Nome: lei
- [] Legislacao - NumArtigo: 15
 - [] - Nome: código penal
- [] OrgaoJulgador - Descricao: segunda câmara criminal

Anote abaixo extrações que você julga que deveriam ter ocorrido (ou marque no acórdão).

.....

.....

.....

DATA:/.../..... RUBRICA:

ANEXO A – EXEMPLO DE ACÓRDÃO

1

APELAÇÃO² CÍVEL Nº 999.999-9³, DA PRIMEIRA VARA⁴ CÍVEL DA COMARCA⁵ DE APUCARANA

APELANTE⁶: BANCO XXXXXXXXXXX XXXXXXXXXXXX S/A⁷

APELADO⁸: MUNICÍPIO DE APUCARANA

RELATORA⁹: DES.^a VILMA RÉGIA RAMOS DE REZENDE

10

CONSTITUCIONAL E TRIBUTÁRIO. EMBARGOS À EXECUÇÃO FISCAL. IMPOSTO SOBRE SERVIÇOS DE QUALQUER NATUREZA (ISSQN). CERTIDÃO DE DÍVIDA ATIVA. ATO ADMINISTRATIVO QUE GOZA DE PRESUNÇÃO DE CERTEZA E LIQUIDEZ. SERVIÇOS BANCÁRIOS. DL¹¹ 406/68. LISTA DE SERVIÇOS ANEXA PELA LC 56/87. INTERPRETAÇÃO AMPLA E ANALÓGICA. INCIDÊNCIA NAS CONTAS "TAXA DE MANUTENÇÃO DE CONTA CORRENTE", "TAXA DE MANUTENÇÃO DE POUPANÇA", "TAXA DE MANUTENÇÃO DE CONTA CORRENTE ATIVA", "OUTROS SERVIÇOS" E "OPERAÇÕES ATIVAS". LEGALIDADE DA APLICAÇÃO DA ALÍQUOTA MÁXIMA DE 5% (CINCO POR CENTO) PREVISTA PELA LC Nº 100/99 ÀS INSTITUIÇÕES FINANCEIRAS. RECURSO¹² PARCIALMENTE PROVIDO.

¹Local de Cabeçalho

²Bonus indicador de Recurso

³Número mascarado para evitar quebra de segredo de justiça

⁴Bonus indicador de Escrivania

⁵Bonus indicador de Comarca

⁶Bonus indicador de Recorrente

⁷Nome mascarado para evitar quebra de segredo de justiça

⁸Bonus indicador de Recorrido

⁹Bonus indicador de Relator

¹⁰Local de seção Ementa

¹¹Bonus indicador de Legislação

¹²Bonus/local indicador de Voto

VISTOS, relatados e discutidos estes autos de Apelação Cível nº 379.178-6, oriundos da Primeira Vara Cível da Comarca de Apucarana, distribuídos para a Primeira Câmara¹⁴ Cível do TRIBUNAL DE JUSTIÇA DO ESTADO DO PARANÁ, em que figuram como Apelante BANCO XXXXXXXXXXXXXXXXXXXXXXX S/A e como Apelado MUNICÍPIO DE APUCARANA.

I - RELATÓRIO

Trata-se de decisão proferida nos autos de Embargos a Execução nº 187/2005, que tramitou perante a Primeira Vara Cível da Comarca de Apucarana, opostos à Execução Fiscal nº. 207/2004 pelo BANCO XXXXXXXXXXXXXXXXXXXXXXX S/A contra o MUNICÍPIO DE APUCARANA, que julgou improcedentes os embargos, sob o fundamento de que (i) os argumentos trazidos pela parte não seriam fortes o suficiente para desconstituir a certidão de dívida ativa, pois carecem de comprovação; (ii) a lista prevista na LC 56/87 admite interpretação extensiva; (iii) as atividades objeto do auto de infração fazem parte do rol constante na Lei¹⁵ Complementar nº. 56/87 e (iv) o que importa é a atividade desenvolvida e não o nome dado pela instituição. A final, condenou o embargante ao pagamento das custas processuais e honorários advocatícios, que fixou em R\$ 3.000,00 (três mil reais), com base no art.¹⁶ 20, §¹⁷ 4º, do Código¹⁸ de Processo Civil.

BANCO XXXXXXXXXXXXXXXXXXXXXXX S/A opôs Embargos de Declaração, os quais foram julgados parcialmente procedentes, apenas para suprir a alegada omissão, declarando a legalidade da incidência da alíquota de 10% (dez por cento) do ISSQN.

BANCO XXXXXXXXXXXXXXXXXXXXXXX S/A postula a reforma da sentença, sustentando: a) a nulidade do título executivo que embasa a Execução Fiscal, pois deixa de especificar a origem do débito fiscal, não havendo como identificar sobre quais atividades estaria a incidir o imposto; b) a nulidade do lançamento que ensejou a inscrição em dívida ativa, pois ao fazê-lo o fisco municipal deixou de mencionar o item da lista de serviços que embasaria a cobrança, ferindo o disposto no artigo 142, do Código Tributário Nacional; c) a ilegalidade da tributação

¹³Local de Relatório

¹⁴Bonus indicador de Órgão Julgador

¹⁵Bonus indicador de Legislação

¹⁶Bonus indicador de Artigo de Legislação

¹⁷Bonus indicador de Parágrafo de Legislação

¹⁸Bonus indicador de Legislação

efetuada pelo Município, pois extrapolou de sua competência ao cobrar ISSQN sobre rubricas contábeis que se referem a atividades não previstas na lista anexa à Lei Complementar, a qual é taxativa; d) a ilegalidade da exação sobre rubricas que registram receitas advindas de atividades acessórias, que não caracterizam prestações de serviços, como a Taxa de Manutenção de Conta Corrente, a Taxa de Manutenção de Poupança, a Taxa de Manutenção de Conta Corrente Ativa, Outros Serviços e Operações Ativas; e) a ilegalidade da alíquota de ISSQN cobrada pelo Município, pois superior à alíquota de 5% (cinco por cento) prevista em Lei Complementar nº. 100/99. O recurso foi recebido em seu efeito apenas devolutivo e contra-arrazoado. O douto Promotor¹⁹ de Justiça Substituto em Segundo Grau NEY ROBERTO ZANLORENZI opinou pelo conhecimento e não provimento do recurso.

É o relatório.

II - VOTO E SUA FUNDAMENTAÇÃO

Presentes os requisitos de admissibilidade, conheço do recurso.

1 - DA VALIDADE DA CERTIDÃO DE DÍVIDA ATIVA E DO LANÇAMENTO EFETUADO PELO MUNICÍPIO O MM. Juiz a quo entendeu por bem rejeitar a preliminar de nulidade da certidão de dívida ativa porque, em se tratando de Execução Fiscal, em que se presume a certeza, liquidez e exigibilidade do título, cabe ao embargante o ônus de desconstituí-la, mediante a apresentação de prova inequívoca.

Segundo o disposto no art. 2º, § 5º, inciso²⁰ III, da Lei nº 6.830/80, o Termo de Inscrição de Dívida Ativa deverá conter "a origem, a natureza e o fundamento legal ou contratual da dívida", estando no mesmo sentido o contido no art. 202, inciso III, do Código Tributário Nacional.

...

Face a alteração parcial do julgado, a readequação dos ônus sucumbenciais é medida que se impõe. Desta forma, condeno o Banco Apelante ao pagamento de 50% (cinquenta por cento) das custas processuais e honorários advocatícios em favor do patrono do MUNICÍPIO DE APUCARANA, que fixo em R\$ 1.500,00 (um mil e quinhentos reais). De igual forma, condeno o MUNICÍPIO Apelado ao pagamento de 50% (cinquenta por cento) das custas processuais e honorários advocatícios ao patrono do Banco Apelante, que fixo em R\$ 1.500,00 (um mil e quinhentos reais).

É como voto.

¹⁹Bonus indicador de Citado

²⁰Bonus indicador de Inciso de Legislação

III - DECISÃO

Diante do exposto, ACORDAM²¹ os Desembargadores integrantes da Primeira Câmara Cível do Tribunal de Justiça do Estado do Paraná, por unanimidade de votos, em dar parcial provimento²² à apelação do BANCO XXXXXXXXXXX XXXXXXXXXXX S/A, para declarar a aplicabilidade do artigo 4º, da Lei Complementar nº 100/99 às instituições financeiras, que fixou em 5% (cinco por cento) a alíquota máxima do ISSQN. A final, condeno o Banco Apelante ao pagamento de 50% (cinquenta por cento) das custas processuais e honorários advocatícios em favor do patrono do Apelado, que fixo em R\$ 1.500,00 (um mil e quinhentos reais) e condeno o MUNICÍPIO DE APUCARANA ao pagamento de 50% (cinquenta por cento) das custas processuais e honorários advocatícios ao patrono do Banco Apelante, que fixo em R\$ 1.500,00 (um mil e quinhentos reais), restando autorizada a compensação nos termos do artigo 21 do Código de Processo Civil e Súmula 306 do SUPERIOR TRIBUNAL DE JUSTIÇA.

Participaram da sessão de julgamento e acompanharam o voto da relatora o Desembargador SÉRGIO RODRIGUES (Presidente) e o Juiz Convocado EDISON DE OLIVEIRA MACEDO FILHO.

Curitiba, 20 de novembro de 2007.

Vilma Régia Ramos de Rezende

DESEMBARGADORA RELATORA

²¹Bonus indicador de local de seção Decisão

²²Bonus indicador de Voto

ANEXO B – URLS DAS PESQUISAS JURISPRUDENCIAIS DOS TRIBUNAIS

Lista de URLs das pesquisas jurisprudenciais consultadas para este trabalho, conforme disponível, na data de 04/01/2010, em http://www.cnj.jus.br/index.php?option=com_content&view=article&id=6540&Itemid=405.

Sigla do Tribunal	URL	Data da consulta
STF	http://www.stf.jus.br/portal/jurisprudencia/pesquisarJurisprudencia.asp	04/01/2010
STJ	http://www.stj.jus.br/SCON/	02/02/2010
TJDFT	http://www.tjdft.jus.br/juris/juris.asp	04/02/2010
TJAC	http://websajju.tjac.jus.br/cjosg/	03/02/2010
TJAL	http://www.jurisprudencia.tj.al.gov.br/busca.aspx	22/02/2010 (indisponível)
TJAP	http://www.tjap.jus.br/	03/02/2010
TJAM	http://consultasaj.tjam.jus.br/cjosg/index.jsp	03/02/2010
TJBA	http://www.tjba.jus.br/	03/02/2010
TJCE	http://www4.tjce.jus.br/sproc2/paginas/Acordao.htm	03/02/2010
TJES	http://www.tjes.jus.br/	05/02/2010
TJGO	http://www.tjgo.jus.br/index.php?sec=consultas&item=decisoes&subitem=jurisprudencia&acao=consultar	05/02/2010
TJMA	http://www.tjma.jus.br/site/cons/jurisp.php	05/02/2010
TJMT	http://www.tjmt.jus.br/servicos/jurisprudencia/Default.aspx	18/02/2010
TJMS	http://www.tjms.jus.br/cjosg/	18/02/2010
TJMG	http://www.tjmg.jus.br/	18/02/2010
TJPA	http://200.217.195.100/consultasProcessuais/jurisprudencia/	18/02/2010
TJPB	http://www.tjpb.jus.br/portal/page/portal/portal.tjpb/home/servicos:indice	24/02/2010 (indisponível)

(continua)

Quadro 16: URLs das pesquisas jurisprudenciais consultadas nos tribunais

Sigla do Tribunal	URL	Data da consulta
TJPR	http://www.tj.pr.gov.br/portal/consultas/consultajurisprudencia.asp?consulta=0	02/02/2010
TJPE	http://www.tjpe.jus.br/jurisprudencia/	22/02/2010
TJPI	http://www.tjpi.jus.br/e-tjpi/home/jurisprudencia	18/02/2010
TJRJ	http://www.tjrj.jus.br/scripts/weblink.mgw?MGWLPN=DIGITAL1A&LAB=XJRPxWEB&PGM=WEBJRP101&PORTAL=1	18/02/2010
TJRN	http://www2.tjrn.jus.br/cjosg/index.jsp	22/02/2010
TJRS	http://www1.tjrs.jus.br/busca/?tb=juris	22/02/2010
TJRO	http://www.tjro.jus.br/cj/faces/jsp/consulta.jsp	22/02/2010
TJRR	http://www.tjrr.jus.br/page/web/tjrr/jurisprudencia	22/02/2010
TJSC	http://app.tjsc.jus.br/jurisprudencia/acpesquisa.action	22/02/2010
TJSP	http://esaj.tj.sp.gov.br/cjsg/consultaCompleta.do	22/02/2010
TJSE	http://jurisprudencia.tjse.jus.br/search?site=Juris&client=juris&output=xml_no_dtd&proxystylesheet=juris&oe=ISO-8859-1&ie=ISO-8859-1&proxycustom=%3CHOME%3E	22/02/2010
TJTO	http://www.tjto.jus.br/	22/02/2010

Quadro 16: URLs das pesquisas jurisprudenciais consultadas nos tribunais