UNISINOS

Programa Interdisciplinar de Pós-Graduação em

Computação Aplicada

Mestrado Acadêmico

Denis Andrei de Araujo

Sistema de Aplicação Unificada de Regras Linguísticas e Ontologias para a Extração de Informações

São Leopoldo, 2013

Denis Andrei de Araujo

SISTEMA DE APLICAÇÃO UNIFICADA DE REGRAS LINGUÍSTICAS E ONTOLOGIAS PARA A EXTRAÇÃO DE INFORMAÇÕES

Dissertação apresentada como requisito parcial para a obtenção do título de Mestre pelo Programa Interdisciplinar de Pós-Graduação em Computação Aplicada da Universidade do Vale do Rio dos Sinos – UNISINOS

Orientador: Prof. Dr. Sandro José Rigo

São Leopoldo
2013

A663s Araujo, Denis Andrei de.
Sistema de aplicação unificada de regras linguísticas e ontologias para a extração de informações / Denis Andrei de Araujo. – 2013.
129 f. : il. ; 30 cm.

Dissertação (mestrado) – Universidade do Vale do Rio dos Sinos, Programa Interdisciplinar de Pós-Graduação em Computação Aplicada, 2013.
"Orientador: Prof. Dr. Sandro José Rigo."

1. Extração da informação. 2. Processamento de linguagem natural (Computação). 3. Ontologias (Recuperação da informação). I. Título.

CDU 004

Dados Internacionais de Catalogação na Publicação (CIP)
(Bibliotecário: Flávio Nunes – CRB 10/1298)

Aos meus pais, pela persistência e dedicação na transmissão de seus valores e princípios.

A minha querida e amada esposa Carolina e meu paciencioso e amoroso filho Daniel, pilares sobre os quais encontro base e equilíbrio para enfrentar os desafios.

Agradecimentos

Agradeço especialmente ao professor e orientador Sandro José Rigo pela constante orientação, pela indicação da direção a ser seguida, pelos aconselhamentos ao longo deste trabalho, pela paciência diante das minhas incessantes dúvidas.

Agradeço à minha querida esposa e companheira Carolina, por sempre ter me apoiado nos momentos de dificuldade, pela paciência nos momentos de angústia e por sempre demonstrar sua crença em minha capacidade, ajudando-me inúmeras vezes a reencontrar o equilíbrio e a força para persistir.

Aos professores do PIPCA pelos ensinamentos, os quais possibilitaram o refinamento de meus conhecimentos na área da Ciência da Computação.

Ao Programa Interdisciplinar de Pós-Graduação em Computação Aplicada da Universidade do Vale do Rio dos Sinos, pela acolhida e oportunidade de realizar esta pesquisa.

Ao professor Mateus Raeder, por ter me concedido a oportunidade de realização do estágio de docência, pela sua paciência e compreensão e pelo espaço de trocas.

À professora Dra. Rove Chishman pela oportunidade e à CAPES por ter me concedido a bolsa de estudos.

Aos meus amigos Me. Daniel Bauermann e Dra. Eliane Schlemmer pelo apoio e confiança no início desta caminhada.

"À vida, que me tem dado tanto."

“O que nos cabe é decidir o que fazer com o tempo que nos é dado.”
R. R. Tolkien

RESUMO

A Extração de Informações é um componente importante do conjunto de ferramentas computacionais que visam a identificação de informação relevante a partir de textos em linguagem natural. Regras de extração de conhecimento, baseadas no tratamento linguístico de aspectos específicos dos documentos textuais, podem contribuir para o alcance de melhores desempenhos nesta tarefa. Este trabalho apresenta um modelo para a Extração da Informação baseada em ontologias, a qual se utiliza de técnicas de Processamento da Linguagem Natural e corpus anotado para a identificação das informações de interesse. São descritos os principais componentes da proposta e apresentado um estudo de caso baseado em documentos jurídicos brasileiros. Os resultados obtidos nos experimentos realizados indicam índices relevantes de acurácia e precisão e boas perspectivas quanto a flexibilidade, expressividade e generalização das regras de extração.

Palavras-chave: Extração da Informação, Ontologias, Processamento da Linguagem Natural

ABSTRACT

Information Extraction is an important part of a broader set of enabling tools to assist on identifying relevant information from natural language texts. Knowledge acquisition rules, based on linguistic treatment of specific aspects of textual documents, can provide an even broader set of possibilities. This work presents a model for addressing Information Extraction from texts based on ontology, which uses Natural Language Processing techniques and annotated corpus to identify relevant information. The main components of the proposal are described and presented a case study based on Brazilian legal documents. The results achieved on experiments indicate relevant accuracy and precision performance and good prospects regarding flexibility, expressiveness and generalization of the extraction rules.

Keywords: Information Extraction, Ontologies, Natural Language Processing.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

DAML	*DARPA Agent Markup Language*
DL	*Description Logic*
EI	*Extração da Informação*
EIBO	*Extração da Informação Baseada em Ontologia*
GATE	*General Architecture for Text Engineering*
JAPE	*Java Annotation Patterns Engine*
LLOD	*Linguistic Linked Open Data*
NIR	*Norme in Rete*
OIL	*Ontology Interchange Language*
OWL	*Ontology WEB Language*
PLN	*Processamento de Linguagem Natural*
POS	*Part of speech*
RDF	*Resource Description Framework*
RDFS	*RDF Schema*
RI	*Recuperação da Informação*
SPARQL	*SPARQL Protocol and RDF Query Language*
SWRL	*Semantic WEB Rule Language*
URI	*Uniform Resource Identifier*
VISL	*Visual Interactive Syntax Learning*
XML	*eXtensible Markup Language*
XSL	*eXtensible Stylesheet Language*
XSLT	*XSL Transformations*

SUMÁRIO

1 INTRODUÇÃO

O aumento na disponibilidade de acesso à informação viabilizado pela Internet é um fenômeno já há algum tempo constatado por especialistas no assunto (MOENS; UYTTENDAELE; DUMORTIER, 1999)(GUARINO, 1997)(RILOFF, 1996)(COWIE; LEHNERT, 1996). Na última década, tem-se cada vez mais evidenciada a necessidade de ferramentas que possibilitem a organização, extração e integração das informações (DEDEK; VOJTAS, 2011)(MAZZEI; RADICIONI; BRIGHI, 2009)(CASTILLO et al., 2003). Estas considerações são ratificadas no contexto da necessária automação do acesso à informação (LESMO et al., 2013)(PALMIRANI et al., 2010)(WIMALASURIYA; DOU, 2010), pois o valor agregado da informação está cada vez menos relacionado à aplicação específica que motivou a sua aquisição, tendendo cada vez mais a depender do grau de reutilização da informação, da possibilidade de sua integração a outras fontes (WIMALASURIYA; DOU, 2009)(LABSKÝ; SVÁTEK; NEKVASIL, 2008).

Desta forma, um aspecto importante passa a ser a identificação de conceitos ou entidades com as quais a informação está relacionada (LACLAVIK et al., 2012) (SITTHISARN; LAU; DEW, 2011). Este referencial de significado precisa ser explicitado e organizado de forma a facilitar que a informação relevante seja recuperada e usada quando necessário (BUITELAAR et al., 2009). Isto é especialmente evidenciado no caso da extração da informação a partir de textos em linguagem natural, quando conceitos específicos precisam ser associados a termos e expressões do texto (KARA et al., 2012)(WYNER; PETERS, 2011) (PALMIRANI et al., 2010)NÉDELLEC; NAZARENKO; BOSSY, 2009).

O processo de análise de documentos, visando realizar a associação entre conceitos específicos de interesse e a forma escrita com que estes foram expressos no texto, chama-se Extração da Informação (COWIE; LEHNERT, 1996). Pode ser aplicado com objetivos específicos, tais como: Reconhecimento de Entidades Nomeadas; Resolução de Correferência; Recuperação da Informação baseada em semântica, entre outras possíveis aplicações (NÉDELLEC; NAZARENKO; BOSSY, 2009). Neste trabalho é descrita uma abordagem para a aplicação da Extração da Informação e apresentando como estudo de caso a sua aplicação sobre o domínio jurídico. A motivação e a escolha de documentos da área jurídica como objeto do estudo de caso são assuntos vistos na seção seguinte.

1.1 Contextualização

A atuação de advogados, juízes, promotores e demais profissionais da área jurídica depende em grande parte da sua capacidade de localizar as informações relevantes para uma correta tomada de decisões (MOENS, 2006). A busca por informações que estejam juridicamente relacionadas a um caso específico ocorre através da pesquisa em documentos jurídicos, tais como leis, jurisprudências[1] e doutrinas[2].

[1] Termo jurídico que refere-se à interpretação da lei baseada em decisões de julgamentos anteriores, que formam uma tradição de decisões sobre causas semelhantes (definição retirada do Dicionário Aulete Digital – acesso em 30. out. 2013)

[2] Na terminologia jurídica, é tido, em sentido lato, como o conjunto de princípios expostos nos livros de Direito, em que se firmam teorias ou se fazem interpretações sobre a ciência jurídica (SILVA, 2010)

Dentre os diversos fatores que influenciam a aplicação correta e apropriada da lei, inclui-se o acesso às informações contidas em casos julgados e nas respectivas leis aplicáveis (REALE, 1977). No âmbito do sistema judiciário brasileiro, a busca por casos semelhantes é denominada de pesquisa de Jurisprudência. Pode-se definir então que a Jurisprudência dos Tribunais constitui-se na busca de casos semelhantes, nos quais objetiva-se identificar uma tendência decisória dos órgãos julgadores.

Vê-se então a importância de dedicar-se esforços no desenvolvimento de sistemas que permitam uma busca ampla, profunda e exaustiva de informações legais para o embasamento das ações dos profissionais desta área. Outro ponto que denota a importância do acesso a informação na área legal é o fato desta área lidar com repositórios de documentos excepcionalmente volumosos. A maior parte destes documentos, inclusive as próprias leis, já se encontram sob o formato digital exatamente com o objetivo de facilitar o seu acesso.

Os repositórios de documentos legais digitalizados resultam em uma quantidade realmente extensa de dados disponibilizados eletronicamente. No entanto, uma parte significativa tende a ser ignorada devido principalmente à sua extensão. A Tabela 1 apresenta, a título de exemplo, a litigiosidade brasileira, a qual permite delinear a dimensão do conjunto de informações jurídicas do Brasil.

Tabela 1: Litigiosidade Brasileira

Justiça	Casos Novos	Casos Pendentes	Total de Processos Baixados	Total de Sentenças/Decisões
Justiça Federal	3.166.766	7.927.287	3.386.186	2.870.562
Justiça do Trabalho	3.316.965	3.278.918	3.454.456	3.454.119
Justiça Estadual	17.743.996	47.960.519	18.476.308	15.827.697

Fonte: Justiça em Números 2010.

O número de litígios gerados, armazenados e manipulados demonstra claramente a magnitude dos repositórios de dados com que os profissionais da área legal precisam lidar. É impossível ao ser humano ler, entender e sintetizar cada um dos documentos que compõe as dezenas de milhões de processos. A perda de informações, causada por esta quantidade exacerbada de dados, afeta diretamente a qualidade da decisão dos profissionais do Direito, sendo então uma área em que se verifica a premência da realização de pesquisas que busquem a otimização do desempenho dos sistemas de Recuperação da Informação (RI) jurídica.

A tendência de concentrar-se o armazenamento dos documentos jurídicos em meios digitais encontra-se também presente no sistema judiciário brasileiro (ALMEIDA Fº, 2010). O projeto brasileiro LEXML [3] (LIMA, 2008), por exemplo, que tem como objetivo a busca de um padrão de representação digital para os documentos jurídicos, está em consonância com outros projetos internacionais, tais como os que estão ocorrendo na Europa no âmbito do Projeto Estrella[4] (BOER et al., 2006)(BOER; WINKELS; VITALI, 2008)(LUPO et al., 2007) e nos Estados Unidos por meio do padrão Oasis LegalXML[5] (HALVORSON, 2002).

[3] http://projeto.lexml.gov.br (acesso em: 25 jul. 2013).
[4] http://www.estrellaproject.org (acesso em: 25 jul. 2013).
[5] http://www.legalxml.com (acesso em: 25 jul. 2013).

Ciente da importância do acesso a estes repositórios digitais para o aprimoramento do Poder Judiciário brasileiro, o Conselho Nacional de Justiça [6] (CNJ) lançou o Programa de Apoio à Pesquisa Jurídica, cujo objetivo é apoiar propostas de desenvolvimento de pesquisas acadêmicas na área de RI jurídica. Neste sentido foi lançado o edital CAPES/CNJ 2010, o qual visa apoiar propostas que promovam e fomentem a realização e a divulgação de pesquisas científicas em áreas de interesse prioritário para o Poder Judiciário nas universidades brasileiras.

Entre as propostas aprovadas no edital, figura o projeto "Tecnologias Semânticas e Sistemas de Recuperação de Informação Jurídica", do Grupo de Pesquisa SEMANTEC [7], cujo o objetivo é desenvolver e implementar um modelo semântico conceitual do domínio jurídico brasileiro, integrando-o a sistemas informatizados de busca e recuperação de documentação jurídica. O sistema de busca e recuperação de documentos jurídicos prevê a proposição de uma arquitetura para um sistema semântico de recuperação de informação baseado na ontologia de domínio produzida pelo projeto.

É neste contexto que este trabalho se insere, visando colaborar na composição da solução para o sistema de RI baseado em semântica proposto no Projeto CNJ Acadêmico. Na definição do escopo deste trabalho, considerou-se como diretrizes: contribuir efetivamente para a definição do sistema de RI semântico previsto no projeto, definindo e implementando integralmente um dos módulos integrantes do sistema; uso da ontologia de domínio jurídico produzida pelo projeto; propor uma solução genérica, que seja também aplicável a outros domínios de conhecimento.

A partir deste direcionamento, iniciou-se a pesquisa por trabalhos científicos que atendessem ao objetivo principal: permitir o desenvolvimento de um sistema de RI baseado em semântica. Entre as diversas soluções encontradas, verificou-se uma estratégia promissora de otimização para o processo de recuperação de documentos baseada em pesquisa semântica de textos: a Extração da Informação (EI).

O objetivo da EI é desenvolver ferramentas e metodologias para identificar, anotar e extrair informações específicas a partir de documentos em linguagem natural. Embora os esforços neste campo não sejam recentes (RILOFF, 1999), a sua importância e necessidade é crescente e está relacionada à grande quantidade de documentos em linguagem natural e às informações textuais relevantes armazenadas em bancos de dados. A análise manual destas bases é impraticável, demandando-se assim o processamento automatizado destes textos. A obtenção de melhores resultados no tratamento das informações armazenadas sob a forma textual através de sistemas de EI possibilitam avanços no desempenho de outros sistemas relacionados, como por exemplo, os sistemas de RI.

Os primeiros sistemas de EI estavam geralmente relacionados a domínios específicos a fim de obter melhores resultados. A frequência e a posição dos termos relacionados ao domínio é um dos principais aspectos considerados nestas abordagens de EI (WIMALASURIYA; DOU, 2009). Baseados nesta premissa, alguns trabalhos bem conhecidos da área (BRUNINGHAUS; ASHLEY, 2001)(JIJKOUN; RIJKE; MUR, 2004) adotam uma abordagem baseada no processamento de texto que desconsidera certas relações

[6] O Conselho Nacional de Justiça é uma instituição pública que visa aperfeiçoar o trabalho do sistema judiciário brasileiro, cuja a missão é contribuir para que a prestação jurisdicional seja realizada com eficiência e efetividade em benefício da sociedade.

[7] Registro CAPES http://dgp.cnpq.br/buscaoperacional/detalhegrupo.jsp?grupo=0009801IL3V0UE (Acesso em: 25 jul. 2013).

e descrições que podem estar disponíveis quando os conhecimentos do domínio e as informações linguísticas são incorporadas ao processo.

Pode-se observar um número expressivo de abordagens que incluem o uso de conhecimento do domínio expressado através de ontologias visando principalmente o aumento da precisão da EI (SITTHISARN; LAU; DEW, 2011)(WIMALASURIYA; DOU, 2010)(SARAVANAN; RAVINDRAN; RAMAN, 2009). Outras, incorporam aspectos linguísticos à solução (LESMO et al., 2013)(WYNER; PETERS, 2011)(SANG; HOFMANN, 2009)(MAZZEI; RADICIONI; BRIGHI, 2009)(LENCI et al., 2007)(AMARDEILH; LAUBLET; MINEL, 2005), tendo por objetivo melhor processar as estruturas complexas da linguagem natural.

Tais abordagens, contudo, não apresentam uma lógica unificada para a implementação do processo de extração, representando uma parte do conhecimento do domínio de forma descritiva na ontologia e outra parte em linguagens de programação procedurais.

Este trabalho apresenta uma metodologia para a EI a partir de textos em linguagem natural que propõe algumas melhorias em relação aos trabalhos acima relacionados. A primeira é o uso intensivo de informações semânticas descritas em ontologias de domínio. A segunda é a incorporação de informações linguísticas obtidas através da análise de documentos do domínio, compondo regras flexíveis e precisas para a extração da informação. Por fim, o uso unificado do sistema lógico inferencial das ontologias para a integração e processamento das informações textuais, de domínio e linguísticos dos documentos.

Embora alguns dos elementos citados possam ser encontrados em outras abordagens (PALMIRANI et al., 2010)(MAZZEI; RADICIONI; BRIGHI, 2009), aplica-se aqui a completa integração destes elementos, visando-se aumentar a flexibilidade da metodologia para lidar de uma forma eficaz com a rica descrição semântica e linguística dos componentes do texto.

O uso do sistema inferencial das ontologias para a elaboração das regras de extração é viabilizado pela representação dos documentos do domínio no modelo de dados POWLA (CHIARCOS, 2012a), um dos componentes do modelo conceitual OLiA [8] (BUYKO; CHIARCOS; PAREJA-LORA, 2008).

O modelo de dados POWLA possibilita a interoperabilidade estrutural e conceitual entre corpora e anotações linguísticas, ambos sendo modelados na linguagem OWL. As ontologias do modelo OLiA2 introduzem um nível de abstração para a representação entre diferentes repositórios linguísticos de referências terminológicas. Informações mais detalhadas sobre os modelos OLiA e POWLA são apresentadas na seção 2.4.

Na próxima seção apresenta-se detalhadamente a questão de pesquisa deste trabalho, definindo-se todos os conceitos envolvidos na sua formulação, buscando assim apresentar uma visão ao mesmo tempo clara e principalmente objetiva da produção científica em vista.

[8] Acrônimo para a expressão em inglês *Ontologies of Linguistic Annotation*.

1.2 Questão de Pesquisa

Diante do contexto exposto na seção anterior e considerando-se que a EI é um elemento essencial para o desenvolvimento de um sistema de RI jurídico baseado em semântica, formulou-se no presente trabalho a seguinte questão de pesquisa:

> *O uso integrado dos mecanismos de inferência da Ontology Web Language (OWL) em conjunto com a ontologia de domínio e as regras de extração permitem a composição de um artefato computacional que possibilite a Extração da Informação com desempenho adequado?*

Os *mecanismos de inferência,* citado na questão de pesquisa, dizem respeito ao uso dos axiomas da linguagem Description Logic (DL) e as regras lógicas da Semantic Web Rule Language (SWRL) para a formalização das regras de extração.

Ontologia de domínio refere-se a um tipo de ontologia quando classificada em relação à sua função. São ontologias caracterizadas pela reutilização dentro do domínio, fornecendo vocabulários sobre os conceitos de um domínio e respectivos relacionamentos, atividades e regras.

A expressão *regras de extração*, no contexto deste trabalho, está relacionada ao conjunto de definições e restrições que permitem a avaliação do conjunto de características das sentenças para a identificação da presença ou não de referências à informações de interesse.

Quanto ao *desempenho adequado*, os processos de EI podem ser avaliados quanto a alguns índices em relação a eficiência do algoritmo implementado. O desempenho da abordagem proposta neste trabalho foi avaliada através da realização de experimentos práticos, pela aplicação da metodologia em corpus do mundo real e a respectiva análise do desempenho do processo de extração.

1.3 Objetivos

O objetivo geral dessa dissertação é propor um processo automático para identificação de conceitos de uma área do conhecimento caracterizada por uma ontologia de domínio e um corpus composto por documentos deste domínio contendo textos em linguagem natural.

Este trabalho descreve uma metodologia para a extração da informação de textos em linguagem natural que identifica traços semânticos nas sentenças através da análise das informações disponibilizada por analisadores automáticos de texto (*parsers*), armazenando as referências identificadas em uma Base de Conhecimento, as quais podem então ser utilizadas por outros sistemas, tais como sistemas de RI baseado em semântica.

O trabalho apresenta dois principais aprimoramentos ao processo de EI: o uso intensivo de ontologias e a utilização unificada do sistema de inferências das ontologias para a formalização das regras de extração. Estas características diferenciadas da metodologia proposta serão resumidamente apresentadas na próxima seção.

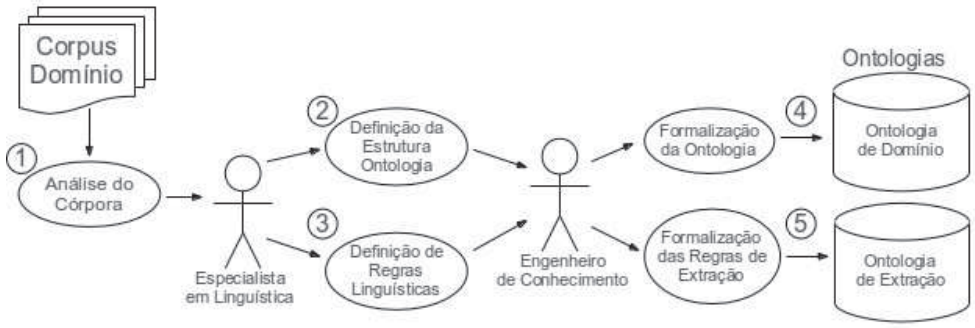
1.4 Metodologia de Trabalho

Esta seção apresenta brevemente os elementos da metodologia deste trabalho. Uma descrição mais detalhada da metodologia e a sua aplicação num estudo de caso é apresentada respectivamente nos capítulos 4 e 5.

A metodologia adotada neste trabalho prevê a coleta de um corpus de domínio a ser submetido a um processo composto de duas etapas. A primeira é denominada de etapa linguística, sendo a seguinte chamada de etapa computacional.

A etapa linguística, como visto na Figura 1, inicia com a análise de um subconjunto de documentos do corpus coletado (1), tendo por objetivo a definição dos conceitos básicos do domínio (2) bem como das regras que serão utilizadas no processo de extração (3). A partir destas definições, são formalizadas a ontologia de domínio (4) e as regras da ontologia de extração (5) (EMBLEY, 2004)(EMBLEY; TAO; LIDDLE, 2005). Ambas as análises são realizadas manualmente pelos respectivos especialistas.

Figura 1: Visão geral dos procedimentos da etapa- linguística.



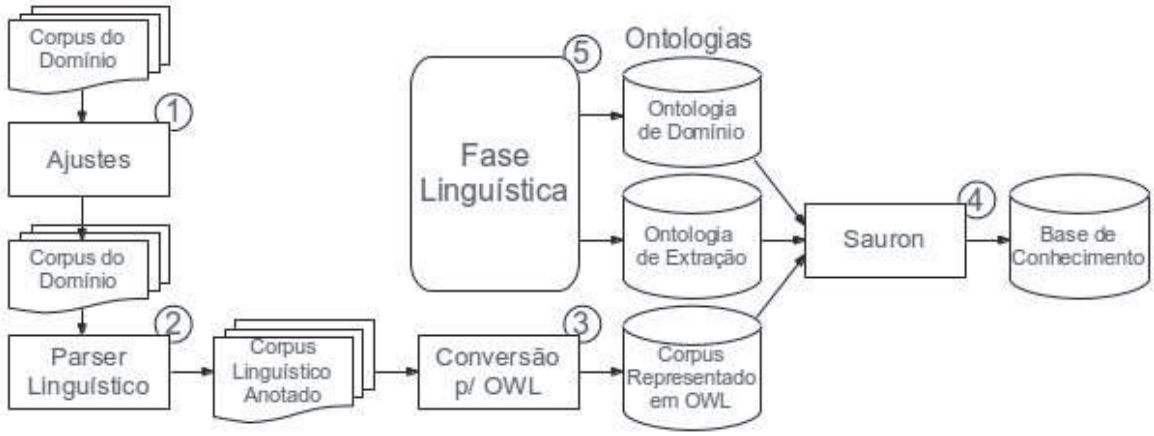
Fonte: elaborado pelo autor

Segue-se a etapa computacional, na qual são utilizadas as ontologias produzidas na etapa anterior para a realização da EI, processo este realizado sobre um novo grupo de documentos do corpus coletado, distintos do utilizado na etapa anterior. Nesta etapa, conforme ilustrado de forma geral na Figura 1, utilizam-se as ontologias de domínio e de extração combinadas com a representação em OWL dos documentos para a efetiva realização do processo de EI.

A representação do corpus em OWL da etapa computacional é resultante de uma sequência de processos automatizados que são aplicados ao corpus do domínio. Primeiramente, são realizados alguns ajustes nos documentos (1), sendo em seguida submetidos a um parser para a geração de documentos anotados (2), os quais serão convertidos para a linguagem OWL (3), quando então estão prontos todos os elementos para a realização do processo de EI.

A extração das informações é realizada pelo Sistema de Aplicação Unificada de Regras e Ontologias (SAURON) (4), uma aplicação desenvolvida no contexto deste trabalho e que tem como objetivo combinar o documento representado em OWL com as ontologias de domínio e de extração geradas na fase linguística (5) para a aplicação das regras de extração e consequente geração da Base de Conhecimento.

Figura 2: Visão geral dos processos da fase computacional.



Fonte: elaborado pelo autor.

A aplicação das regras se dá através da ativação do sistema de inferência ontológico, o qual avalia os axiomas e regras lógicas formalizados em cada uma das ontologias que compõem a solução para a geração da Base de Conhecimento, esta também representada no formato OWL. Após a realização dos experimentos, foram avaliados os desempenhos da abordagem aqui sugerida, através da medição da efetividade do processo de extração.

1.5 Organização do Documento

O texto está organizado como segue: no capítulo 2 encontra-se a fundamentação teórica utilizada para o desenvolvimento da abordagem, sendo no capítulo 3 descritos e analisados os trabalhos relacionados à área de pesquisa deste trabalho. No capítulo 4, apresenta-se o modelo proposto, sendo expostos e comentados no capítulo 5 os resultados obtidos nos experimentos realizados em um estudo de caso. Por fim, no capítulo 6, são apresentadas as conclusões e contribuições científicas deste trabalho, bem como vislumbradas as possibilidades da continuidade da pesquisa aqui iniciada em trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta ao leitor uma síntese dos fundamentos teóricos que embasaram a metodologia aqui descrita. São revisadas as principais áreas de conhecimento pertinentes à abordagem aqui proposta, tomando-se por objetivo situar o leitor quanto às teorias e técnicas adotadas para formatar a solução implementada neste trabalho.

Inicia-se pelos conceitos básicos sobre ontologias do ponto de vista da Representação do Conhecimento, passando-se pela linguagem de representação do conhecimento conhecida por Lógica de Descrição (DL) e culminando com uma descrição detalhada da linguagem de representação de ontologias OWL, apresentando-se as suas características principais, a sua terminologia específica e, por fim, explanando-se sobre a linguagem SWRL, a qual implementa a inferência baseada em regras da OWL.

Posteriormente, são vistos os fundamentos do Processamento de Linguagem Natural (PLN), uma subárea da Inteligência Artificial e da Linguística que estuda os problemas da geração e compreensão automática das linguagens utilizadas para comunicação pelo ser humano, descrevendo-se de forma resumida os seus níveis de processamento, as categorias e as principais aplicações desta técnica.

Após, apresenta-se os fundamentos do modelo conceitual OLiA, um conjunto de ontologias que visa a interoperacionalização de anotações linguísticas, aprofundando-se em um de seus componentes, o modelo de dados POWLA, este um elemento essencial para a implementação deste trabalho.

Por fim, são apresentados os principais conceitos da Extração da Informação, descrevendo-se os seus conceitos básicos e os objetivos da área, situando-a funcionalmente e elencando-se as principais abordagens existentes. Explicita-se também a relação entre EI, ontologias e linguística, para então fechar o capítulo explanando-se especificamente sobre a Extração da Informação Baseada em Ontologias, abordagem na qual baseia-se a implementação deste trabalho.

2.1 Ontologias

A Representação do Conhecimento (RC) é uma área da Inteligência Artificial que visa representar simbolicamente o conhecimento de forma a permitir o seu processamento computacional, possibilitando inclusive a produção de novos conhecimentos inferidos a partir das informações armazenadas (SOWA, 2000).

A pesquisa na área de RC pressupõe a análise de como implementar modelos computacionais de raciocínio acurados e efetivos, bem como definir o conjunto de símbolos necessários para a representação dos fatos de um determinado domínio para que sistemas baseado em conhecimento possam utilizar estes modelos para a produção coerente de novos fatos.

As pesquisas em RC obtiveram seu auge de popularidade na década de 70. As abordagens desenvolvidas nesta época são genericamente divididas em duas categorias: aquelas baseadas em representações sem o uso da lógica, tais como as redes semânticas, por exemplo e as com formalismo baseado em lógica, originadas da intuição de que o cálculo de

predicado poderia ser utilizado para representar fatos do mundo real de forma não ambígua (NARDI; BRACHMAN, 2003).

A RC não baseada em lógica baseia-se geralmente sobre fundamentos mais cognitivos, sendo as suas representações estruturadas sob a forma de redes ou baseadas em regras. As bases teóricas deste tipo de RC derivaram-se de experimentos baseados na capacidade de recuperação de informações da memória humana e na capacidade humana de execução de tarefas (NARDI; BRACHMAN, 2003). Muito embora tais experimentos fossem sempre muito específicos, os seus resultados eram vistos como aplicáveis em diferentes domínios e para diferentes tipos de problemas.

Nas abordagens baseadas em lógica, as linguagens utilizadas para a representação do conhecimento são normalmente variações do cálculo de predicados de primeira ordem (NARDI; BRACHMAN, 2003), o que proporciona desde o princípio aplicações intrinsecamente generalizadas, complementadas pela inferência lógica para verificação de consequências.

A ontologia é um exemplo clássico de linguagem de representação do conhecimento baseada em lógica e que muito tem chamado a atenção da comunidade científica das áreas de Inteligência Artificial e Ciência da Informação (BREUKER; HOEKSTRA, 2004) (AMARDEILH; LAUBLET; MINEL, 2005)(BUYKO; CHIARCOS; PAREJA-LORA, 2008) (KARA et al., 2012).

A Ontologia é o ramo da Filosofia que estuda as coisas que existem, os tipos e estruturas de objetos, propriedades, eventos, processos e relações em cada área da realidade. O termo “ontologia” é muitas vezes utilizados pelos filósofos como um sinônimo para “metafísico”. Era um termo utilizado pelos primeiros estudiosos de Aristóteles para referir-se ao que ele chamava de “primeira filosofia”.

Diferentemente do significado dado pela filosofia, o conceito de ontologia para a Inteligência Artificial (IA) está mais próximo da epistemologia, que é “um campo da filosofia que lida com a natureza e as fontes do conhecimento” (NUTTER, 1987).

A definição mais comumente utilizada na IA é que ontologia é “uma especificação de uma conceitualização” (GRUBBER, 1993). Ou seja, uma ontologia é uma descrição de conceitos e relacionamentos que um indivíduo ou comunidade de indivíduos podem formular para si (MOMMERS, 2001).

Borst (1997) redefine ontologia como sendo uma “especificação formal de uma conceitualização compartilhada”, incluindo o requisito de que uma ontologia deve obrigatoriamente expressar uma visão compartilhada, um consenso, em vez de uma visão individual.

A estruturação de uma ontologia se dá por um conjunto de definições em um vocabulário formal. Muito embora não seja esta a única forma de especificar uma conceitualização, esta abordagem traz consigo algumas vantagens aos sistemas de IA, como por exemplo a semântica independente de leitor e contexto (GRUBBER, 1992).

Quando o conhecimento de um domínio é representado em um formalismo declarativo, o conjunto de objetos que podem ser representados é chamado de universo de discurso (GRUBBER, 1995). Este conjunto de objetos e os respectivos relacionamentos são refletidos no vocabulário representacional, que são utilizados para a representação do conhecimento em sistemas computacionais.

Assim, no contexto da IA, uma ontologia pode ser descrita pela definição de um conjunto de termos representativos. Os termos podem associar os nomes de entidades do universo de discurso (classes, relações, funções ou outros objetos) a textos que descrevem o que os nomes significam, enquanto axiomas formais impõem restrições à interpretação dos termos, também verificando o seu bom uso.

Formalmente, uma ontologia é a declaração de uma teoria lógica (GRUBER, 1995). Na ciência da computação, uma ontologia é um tipo especial de objeto de informação e um artefato computacional (GUARINO, 2009).

As ontologias computacionais são um meio pelo qual pode-se modelar formalmente a estrutura de um domínio, identificando as entidades e relacionamentos relevantes que o compõem, desde que sejam úteis para o propósito para o qual a ontologia será utilizada. A modelagem deve ser especificada em uma linguagem não ambígua, computacionalmente processável e humanamente compreensível.

O uso efetivo da ontologia depende, além da linguagem descritiva, também da sua capacidade de inferência. A inferência é importante tanto para se garantir a qualidade e a integridade de uma ontologia quanto para melhor se aproveitar a flexibilidade estrutural com que os seus dados podem ser armazenados e relacionados. Os benefícios da inferência podem ser aproveitados em diversos momentos do ciclo de vida da ontologia.

Durante a fase de projeto, por exemplo, a inferência pode ser utilizada para verificar se não existem conceitos contraditórios ou se a organização dos elementos e relações está coerente com o domínio sendo modelado. Na fase de uso efetivo, é possível utilizar-se a capacidade de generalização ou especialização inferencial para expandir ou restringir conceitos, dependendo dos objetivos da aplicação.

As ontologias representam uma base de conhecimentos que pode ser explorada para melhorar o desempenho das técnicas utilizadas nas várias áreas da ciência da computação que lidam com o processamento de textos, sendo foco específico deste trabalho o seu emprego na área da Extração da Informação.

2.1.1 Lógica de Descrição

A linguagem de representação de conhecimento conhecida por Lógica de Descrição (DL[9]) (BAADER; SATTLER, 2001)(CALVANESE et al., 2001) provê uma semântica bem definida e um poderoso conjunto de ferramentas de inferência que possibilitam a construção, integração e evolução de ontologias de alta qualidade (BAADER; HORROCKS; SATTLER, 2009).

A DL é uma família de linguagens utilizada para a representação estruturada e formalizada do conhecimento de um domínio de aplicação. O nome *lógica de descrição* origina-se do fato de que noções importantes do domínio são explicitadas pela descrição conceitual, ou seja, expressões que são construídas a partir de conceitos atômicos (predicados unários) e regras atômicas (predicados binários), os quais utilizam os construtos de conceitos e regras fornecidos pela implementação da DL. Diferentemente dos seus predecessores, tais como as redes semânticas, a DL tem uma semântica formal bem definida e baseada em lógica (BAADER; HORROCKS; SATTLER, 2009). O exemplo abaixo, retirado de Baader, Horrocks e Sattler (2009), ilustra o uso da linguagem DL para descrever o conceito "Um

[9] Acrônimo para a expressão em inglês *Description Logic*.

homem que é casado com uma doutora e que tem pelo menos cinco filhos, todos eles professores":

$$Humano \sqcap \neg Fêmea \sqcap \exists\ casado.Doutor \sqcap (\geq 5 temFilho\ ) \sqcap \forall\ \ temFilho.Professor$$

No exemplo acima são utilizados os construtos booleanos *conjunção* ($\sqcap$) e negação ($\neg$), que são interpretados respectivamente como intersecção e complemento de conjuntos. Além dos booleanos, são utilizado também os construtos de *restrição existencial* ($\exists$), de *restrição de valor* ($\forall$) e de *restrição numérica* ($\geq$).

Um indivíduo, *João* por exemplo, pertence ao conjunto $\exists casado.Doutor$ se existir outro indivíduo com o qual *João* seja casado (ou seja, alguém com quem João tenha uma relação *casado*) e que seja um doutor (pertença ao conjunto Doutor). Este indivíduo *João* pertence ao conjunto *(≥5 temFilho)* se tiver 5 ou mais filhos. Analogamente, pertence ao conjunto $\forall temFilho.Professor$ se todos os filhos do indivíduo (todos os indivíduos com os quais tenha uma relação *temFilho*) forem professores.

Além deste formalismo descritivo, a linguagem DL conta com um formalismo terminológico e assertivo. Axiomas terminológicos poderiam ser utilizados, por exemplo, para referenciar descrições complexas de forma abreviada, através de um nome. Seria possível em DL denominar a relação apresentada no exemplo do parágrafo anterior com o termo *HomemFeliz*. Pode-se também definir restrições mais genéricas, explorando-se mais profundamente a expressividade dos formalismos terminológicos. Para referir-se ao conjunto de axiomas terminológicos da DL convencionou-se o uso do termo TBox.

O formalismo assertivo da DL é utilizado para a definição das propriedades dos indivíduos. Recorrendo-se novamente ao exemplo de Baader, Horrocks e Sattler (2009), pode-se declarar os seguintes axiomas assertivos:

$$Professor(Maria), temFilho(João, Maria\ )$$

Os axiomas acima definem que Maria é professora e é uma das filhas de João. Ao conjunto de axiomas assertivos dá-se o nome de ABox, sendo chamados de indivíduos os elementos nela referenciados.

A escolha da DL como semântica formal para ontologias ganhou força com a sua indicação para compor a Ontology Web Language (OWL) na recomendação proposta pelo grupo OWL-WG do Consórcio W3C. A indicação da DL como base da semântica formal da OWL tornou-se uma recomendação formal da W3C em outubro de 2004.

A OWL é na verdade uma família de linguagens para representação do conhecimento utilizada para a criação de ontologias, caracterizada pela semântica formal definida e pela serialização baseada no padrão RDF/XML (BRICKLEY; GUHA, 2004). Oficialmente recomendada pela W3C para ser utilizada na escrita de ontologias para a futura Web Semântica (BERNERS-LEE; HENDLER; LASSILA, 2001), a OWL despertou a atenção do mundo acadêmico e comercial. A próxima seção apresenta informações sobre a origem e principais características da linguagem OWL.

2.1.2 A Linguagem para representação de Ontologias na Web OWL

Pode-se observar que o desenvolvimento das ontologias tem um longo histórico na Filosofia e na Ciência da Computação. Nota-se, principalmente a partir da década de 90, um número crescente de pesquisas para explorar o uso de ontologias na representação do

conhecimento, visando principalmente seu aproveitamento na *World Wide Web* (BAADER; HORROCKS; SATTLER, 2009).

Em 2000, James Hendler liderou o desenvolvimento da linguagem DAML [10] (HORROCKS, 2002) em um projeto financiado pela agência de defesa norte americana DARPA (Defense Advanced Research Projects Agency). Em março de 2001 houve a fusão da DAML com a Ontology Inference Language (OIL), a linguagem para escrita de ontologias desenvolvida no projeto OntoKnowledge do programa europeu denominado Information Society Technology (IST), originando daí a linguagem DAML+OIL. A linguagem OWL iniciou como uma pesquisa de revisão da DAML+OIL com vistas a ser o padrão de fato para a web semântica.

A sintaxe da linguagem OWL, como já visto na seção anterior, é baseada no padrão RDF Schema (RDFS), mas o seu projeto baseou-se fortemente na expressiva linguagem lógica de descrição SHIQ (HORROCKS; SATTLER; TOBIES, 2000), usada para descrever explícita e formalmente as conceitualizações de modelos de domínios em ontologias (BAADER; HORROCKS; SATTLER, 2009).

A escolha do RDFS como padrão sintático e de serialização para a OWL tem alguns efeitos colaterais, devido às suas limitações quanto à expressividade, pois o RDFS na sua forma pura permite a representação de somente alguns conceitos ontológicos. Os princípios de modelagem do RDFS visam primariamente a organização de vocabulários em relacionamentos de subclasses e subpropriedades, restrições de domínios e intervalos, bem como instanciação de classes. Resulta daí a inadequação do padrão RDFS como linguagem para descrição de ontologias, uma vez que a sua expressividade não é suficiente para representar conhecimento como requerido pela ontologia. O OWL estende os conceitos de classes e propriedades do RDFS, adicionando primitivas para suportar a expressividade necessária para a descrição de ontologias.

No entanto, a ideia de estender o RDFS apresenta alguns percalços na prática, principalmente no que tange à relação entre poder de expressividade e eficiência computacional de inferência. Primitivas do RDFS, tais como rdfs:Class (conjunto de todas as classes) e rdfs:Property (conjunto de todas as propriedades), podem levar a situações computacionais incontroláveis se implementadas conforme a expressividade original do RDFS (ANTONIOU; HARMELEN, 2009).

Tendo em vista esta questão, o OWL-WG sugeriu três espécies de linguagens para a versão 1.1 do OWL, cada uma delas oferecendo vantagens para cenários específicos de aplicação. Estas sugestões tornaram-se uma recomendação formal do W3C em outubro de 2004, definindo três sublinguagens: OWL Lite, direcionada para usuários com necessidades básicas de classificação hierárquica e restrições simples; OWL DL para aplicações que necessitem a maior expressividade possível, mantendo-se a decidibilidade computacional e a possibilidade de uso de algoritmos de inferência; OWL Full, uma versão baseada em uma semântica diferenciada, projetada para manter alguma compatibilidade com o padrão RDFS.

Em outubro de 2009 o W3C anunciou a OWL versão 2 (OWL-WG, 2009), com várias extensões e novas características para a linguagem. Nesta versão, são descritos três subconjuntos sintáticos da linguagem OWL (MOTIK et al., 2009), chamados de perfis: a OWL 2 EL, que habilita algoritmos de tempo polinomial para todos os processos de inferência padrão, indicado para aplicações onde a ontologia é muito extensa e o poder de

[10] Acrônimo para a expressão em inglês *DARPA Agent Markup Language*.

expressividade pode ser reduzido a fim de obter-se garantia de desempenho; a OWL 2 QL, que habilita consultas conjuntivas (KOLAITIS, 2000) respondidas dentro do espaço de memória logarítmico usando a tecnologia padrão de banco de dados relacional; por último, a OWL 2 RL, que habilita a implementação de algoritmos de inferência em tempo polinomial usando tecnologia de banco de dados estendido para regras, operando diretamente sobre as triplas RDF, sendo este perfil indicado para ontologias médias mas com um número de indivíduos muito extenso.

Com relação ao armazenamento em arquivos, foi definido pelo W3C que o padrão RDFS é a sintaxe concreta oficial e obrigatória para o OWL (OWL-WG, 2009), devendo esta ser suportada por todos os sistemas que manipulam ontologias OWL. No entanto, outras sintaxes podem ser adicionalmente utilizadas. Na Tabela 2 são listadas as sintaxes oficialmente recomendadas pelo Consórcio W3C.

Tabela 2: Sintaxes para OWL.

Nome	**Status**	**Propósito**
RDF/XML	Obrigatória	Intercâmbio (leitura/escrita por qualquer sistema em conformidade com OWL 2).
OWL/XML	Opcional	Facilita o processamento pelo uso de ferramentas XML.
Funcional	Opcional	Facilita a visão das estruturas formais da ontologia.
Manchester	Opcional	Facilita a leitura/escrita de ontologias baseadas em DL.
Turtle	Opcional (não originada do grupo OWL-WG)	Facilita a leitura/escrita de triplas RDF.

Fonte: OWL-WG (2009)

Quanto à semântica, a especificação do OWL 2 define a sua estrutura abstrata, mas não define seu significado. São utilizadas a *Semântica Direta (MOTIK; PATEL-SCHNEIDER; GRAU, 2009)* e a *Semântica Baseada em RDF* (SCHNEIDER, 2009) como formas para expressar o significado na linguagem OWL 2, ambas ligadas uma a outra através de um teorema de correspondência. Estas duas semânticas são utilizadas pelos algoritmos de inferência para verificação da consistência das classes, subsunção e também para consulta de instâncias.

A *Semântica Direta* atribui significado à estrutura da ontologia, compatibilizando-se assim com o modelo teórico semântico da lógica de descrição SROIQ (HORROCKS; KUTZ; SATTLER, 2006). A *Semântica Baseada em RDF* atribui significado indiretamente à estrutura da ontologia via os grafos RDF, sendo totalmente compatível com a *Semântica RDF* (HAYES, 2004) e estendendo as condições semânticas definidas para o RDF.

O teorema de correspondência define que dada uma ontologia OWL 2 DL, inferências deduzidas via *Semântica Direta* deverão continuar válidas se a ontologia for mapeada em um grafo RDF e interpretada com a *Semântica Baseada em RDF*.

2.1.3 Terminologia OWL

Na prática, uma ontologia escrita na linguagem OWL pode ser vista como um conjunto de axiomas que provêm uma lógica explícita de asserções especificamente sobre três elementos: Classes, Instâncias e Propriedades. Novos fatos, implicitamente descritos na ontologia, podem ser inferidos pelo uso de um sistema para raciocínio auxiliar chamado *reasoner*. O *reasoner* analisa os conceitos e relações explicitados na ontologia, concluindo novos relacionamentos entre os seus conceitos e instâncias.

Uma *Classe* é uma coleção de objetos, correspondendo ao elemento denominado *conceito* na DL. Uma classe pode ter *Instâncias*, que corresponde ao conceito de *indivíduos* da DL. Uma *Instância* pode pertencer à nenhuma, uma ou mais *Classes*. Uma *Classe* pode ser uma *subclasse*, herdando as características das suas *superclasses* superiores. Esta herança corresponde à *subsunção lógica* e ao conceito *inclusão* da DL denotado pelo símbolo $\sqsubseteq$. Todas as classes de uma ontologia OWL são subclasses de *owl:Thing* e têm *owl:Nothing* como subclasse.

As *Propriedades* do OWL são uma relação binária direcionada que especifica as características das *Classes* e corresponde ao conceito de *Papel* na lógica DL. São atributos das *Instâncias*, ora atribuindo-lhes dados valorados, ora associando-os a outras *Instâncias*. Suportam características lógicas de transitividade, simetria, inversão e funcionalidade. Podem ser do tipo *Dado* (*Data Property*) ou *Objeto* (*Object Property*). As *Propriedades* tipo *Dado* descrevem relações binárias que ligam um *Indivíduo* a um dado tipificado, como por exemplo um xsd:integer para valores numéricos inteiros ou xsd:string para sequências de caracteres. As *Propriedades* tipo *Objeto* descrevem uma relação entre dois *Indivíduos* da ontologia OWL.

Uma limitação da linguagem OWL é a impossibilidade de representar-se através das *Propriedades* as relações com aridade maior que um, suportando única e exclusivamente relações binárias. Esta limitação pode ser contornada adotando-se alguns padrões na implementação da ontologia, como os propostos por Noy e Rector (2006). Todavia, é possível implementar relações n-árias de uma forma mais natural, elegante, legível e flexível, utilizando-se a representação de conhecimento baseado em regras. A próxima seção apresenta alguns detalhes sobre uma linguagem baseada em regras proposta para a *Web Semântica* submetida ao W3C para compor a linguagem OWL.

2.1.4 A Linguagem baseada em Regras para a Web Semântica - SWRL

Tendo em vista algumas limitações da linguagem OWL, como por exemplo a dificuldade em representar relacionamentos com mais de dois participantes, mas também devido à importância de sistemas baseados em regras para o desenvolvimento de aplicações comerciais, a busca da integração da linguagem OWL com a representação de conhecimentos e inferência baseado em regras despertou o interesse acadêmico (HITZLER; PARSIA, 2009).

Tem-se o desenvolvimento de uma proposta de linguagem baseada em regras para o OWL, referenciada pelo nome *OWL Rule Language* (HORROCKS; PATEL-SCHNEIDER, 2004). Posteriormente rebatizada para Semantic Web Rule Language, tornou-se uma proposta concreta de linguagem baseada em regras para a Web Semântica (HORROCKS et al., 2004) que combina a sintaxe concreta das sublinguagens OWL DL e OWL Lite com a sintaxe da linguagem RuleML[11].

[11]Acrônimo para "Rule Markup Language", a RuleML é uma linguagem de marcação desenvolvida para expressar regras em XML (BRAY et al., 2006) para dedução, reescrita e demais processos inferenciais/transformacionais.

A linguagem SWRL é a união irrestrita da OWL DL com a lógica binária de Horn presente no Datalog, tradicional sistema de banco de dados lógico baseado em regras. Resulta daí um formalismo muito expressivo, porém indecidível (HITZLER; PARSIA, 2009). A decidibilidade no SWRL é alcançada através da imposição de condições de segurança sobre as suas regras, referenciadas pelo nome de *DL-Safety*. As regras SWRL são chamadas de *regras SWRL DL-Safe* ou simplesmente *regras DL-Safe*. A decidibilidade das regras *DL-Safety* foram estabelecidas em Motik, Sattler e Studer (2005) e posteriormente reelaboradas no trabalho desenvolvido em Motik (2006).

As regras propostas na linguagem SWRL seguem a forma de uma implicação entre um antecedente (corpo) e um consequente (cabeçalho). O significado pretendido pode ser lido como: se as condições especificadas no antecedente são verdadeiras, então as condições definidas no consequente (devem) ser verdadeiras também.

Tanto o antecedente (corpo) quanto o consequente (cabeçalho) podem conter zero ou mais *átomos*. Um antecedente vazio é verdadeiro por definição, sendo então o consequente tomado como verdadeiro também. Um consequente vazio é tomado por definição como falso, resultando em um antecedente também falso. Múltiplos *átomos* são interpretados como se estivessem conectados por uma *conjunção lógica*.

No contexto da SWRL, átomos podem estar representados pelos predicados *C(x), P(x,y), sameAs(x,y)* ou *differentFrom(x,y)*, no qual *C* é uma descrição de uma classe OWL, *P* é uma *Propriedade* e *x,y* podem ser *variáveis*, *instâncias* ou valores de dados. Os predicados assertivos *sameAs(x,y)* e *differentFrom(x,y)* definem condições de igualdade ou desigualdade entre os elementos *x* e *y*.

Como já salientado nesta seção, a linguagem OWL não é capaz de expressar toda e qualquer relação existente, restringindo-se unicamente às relações binárias. Uma relação não binária simples, como por exemplo *filho de pais casados*, não pode ser representada diretamente em OWL porque não há como expressar que um *Indivíduo* (filho) relaciona-se com uma relação entre *Indivíduos* (casados) (KUBA, 2012).

Contudo, com o suporte a linguagem SWRL na OWL, torna-se simples a implementação deste relacionamento pela definição de uma regra definindo que um indivíduo *x* da *Classe Pessoa*, cujos pais *y* e *z* tenham um relacionamento *temCônjuge*, pertence à classe *FilhoDePaisCasados*, ficando assim definido a regra:

Pessoa(?x), temProgenitor(?x , ?y), temProgenitor(?x , ?z), temCônjuge(?y , ?z) -> FilhoDePaisCasados(?x)

Neste exemplo são apresentados os seguintes *átomos*/*predicados* no *antecedente*/*corpo* da regra: a classe *Pessoa* e as *Propriedades Objeto temProgenitor* e *temCônjuge*. O *consequente/cabeçalho* da regra define uma nova *Classe* da ontologia, definida pelo termo *FilhoDePaisCasados*. Ao utilizar-se um *reasoner* que suporte as regras da linguagem SWRL, será inferido que *x* pertence à *Classe FilhoDePaisCasados*, representando assim a relação pretendida.

As regras SWRL podem conter como predicados *Classes* e *Propriedades* simples ou com expressões de restrição arbitrárias, restrições de intervalos de dados, predicados

pré-definidos[12] nucleares ou personalizáveis. Para melhor entender o uso destes predicados, são apresentados alguns exemplos, retirados do trabalho de Kuba (2012).

Pessoa(?x), temFilho min 1 Pessoa(?x) -> Progenitor(?x)

A regra acima define que *Indivíduos* (?*x*) que pertençam à classe *Pessoa* e que tenham pelo menos uma relação *temFilho* com outro *Indivíduo* pertencem à classe *Progenitor*. Ou seja, pessoas que tem pelo menos um filho são progenitoras. É importante ressaltar que o exemplo apresentado tem objetivo puramente didático, pois poderia ser definido sem a utilização de regras SWRL, uma vez que apresenta somente a explicitação de uma restrição de uma regra binária, plenamente representável em DL.

A restrição de intervalos de dados é utilizada no exemplo apresentado abaixo, também retirado do tutorial de Kuba (2012):

Pessoa(?p), integer[>= 18, <= 65](?idade), temIdade(?p, ?idade) -> temIdadeMotorista(?p, true)

A restrição de intervalo de dados é satisfeita quando a variável *idade* contiver um valor inteiro entre 18 e 65. Este exemplo é um pouco mais complexo, pois apresenta a utilização dos *facets*[13] da linguagem OWL. Um exemplo mais simples seria a utilização das restrições de dados para verificação pura e simples do tipo da variável, aplicando-se por exemplo a restrição "integer(?idade)", sem o uso dos *facets*.

A proposição da SWRL submetida a W3C define alguns predicados que manipulam os valores das propriedades de dados, chamados de predicados nucleares SWRL pré-definidos, os quais são suportados por alguns reasoners, tais como Pellet [14] e HermiT [15]. Os predicados pré-definidos são precedidos com o descritor de espaço de nomes[16] "swlrb:", por exemplo:

Pessoa(?p), temIdade(?p, ?idade), swrlb:greaterThan(?idade, 18) -> Adulto(?p)

A regra SWLR acima define que as pessoas que tem idade acima de dezoito anos pertencem à classe *Adulto*, utilizando o predicado pré-definido *swrlb:greaterThan*. Existem algumas restrições para o uso dos predicados pré-definidos, as quais estão diretamente relacionadas ao *reasoner* sendo utilizado.

Para aqueles casos em que os predicados pré-definidos não atendem as necessidades de expressão de regras, é possível definir predicados personalizados, como no exemplo:

Pessoa(?p), nascidaAno(?p, ?ano), meu:anoAtual(?anoatual),
swrlb:subtract(?idade, ?anoatual, ?ano) -> temIdade(?p, ?idade)

A regra acima faz uso do predicado personalizado *meu:anoAtual* para atribuir o valor inteiro referente ao ano atual à variável *anoatual*, utilizando esta variável para calcular a idade da pessoa *p*, subtraindo o *ano* de nascimento do *anoatual*. Observe-se ainda que a regra acima resulta não em uma nova classe, mas sim uma nova propriedade de dados chamada *temIdade*.

Embora a inclusão da SWRL visivelmente agregue uma maior expressividade ao OWL, observa-se na prática uma dificuldade para a implementação de ontologias baseadas

[12] O termo "pré-definido" é uma tradução livre para a expressão em inglês "built-in" (nota do autor).
[13] http://www.w3.org/TR/owl2-syntax/#Datatype_Maps
[14] <http://clarkparsia.com/pellet>. Acesso em: 11 jul. 2013.
[15] <http://www.hermit-reasoner.com>. Acesso em: 11 jul. 2013.
[16] A expressão "descritor de espaço de nomes" é uma tradução livre para o termo em inglês "namespace".

em regras. Mesmo sendo uma padronização formalmente definida pela W3C, ao implementar-se as regras utilizando-se recursos mais avançados da linguagem SWRL, como por exemplo os predicados personalizados, nota-se uma falta de documentação e real padronização. Até mesmo a sintaxe pode variar, dependendo do reasoner utilizado, o que afronta os princípios básicos de compartilhamento e reutilização das ontologias.

No entanto, as restrições resultantes da falta de padronização somente são percebidas em casos muito específicos, ao aplicar-se os recursos avançados dos predicados personalizados.

2.2 Processamento da Linguagem Natural

A linguagem é uma forma de comunicação que é efetivada pela troca de mensagens representadas por uma combinação específica de sinais gráficos ou acústicos entre comunicantes que compartilham um senso comum de conhecimento. A linguagem possui níveis de análise: a pragmática, que engloba o meio em que vivem os participantes da comunicação; a semântica, que são as relações entre as expressões da linguagem e os seus significados; e, por fim, o nível de sintaxe que examina as propriedades e estruturas da linguagem. O léxico e a morfologia são subníveis do nível sintático e dizem respeito às palavras e sua formação (flexões, derivações e composição) (LIDDY, 2003).

Ambiguidades podem ser produzidas em cada um dos níveis da linguagem, as quais podem ser resolvidas pelos níveis subsequentes. No entanto, algumas ambiguidades somente podem ser resolvidas pelo conhecimento do contexto em que a frase está inserida. É senso comum que os seres humanos normalmente utilizam todos estes níveis de análise linguística para a desambiguação do significado.

O Processamento de Linguagem Natural (PLN) refere-se a abordagem computadorizada para o tratamento das ambiguidades utilizando uma combinação de níveis de análise linguística, baseada em um conjunto específico de teorias e tecnologias. São os diferentes conjuntos de níveis de análise linguística que cada sistema de PLN utiliza que os diferenciam entre si. Além do conjunto de níveis utilizados, os sistemas de PLN podem diferenciar-se também pelo tipo de análise que implementam: superficial ou aprofundada (LIDDY, 2003).

O PLN é considerado uma disciplina da Inteligência Artificial, uma vez que trata do processamento da linguagem humana. Geralmente utilizado como ferramenta de apoio para a execução de tarefas ou para o desenvolvimento de aplicações específicas, está dividido em duas áreas: processamento da linguagem e geração da linguagem. O processamento refere-se à análise da linguagem para a produção de uma representação do significado, enquanto que a geração trata da produção da linguagem a partir da representação. Devido ao escopo deste trabalho, será vista somente a análise da linguagem natural, uma vez que é esta a área relevante para a metodologia de EI aqui sugerida.

2.2.1 Os Níveis de Processamento da Linguagem

Uma abordagem interessante para explicar o que realmente acontece dentro de um sistema de PLN é analisar os níveis de análise da linguagem que o sistema implementa.

Pesquisas na área da psicolinguística sugerem que o processamento da linguagem é altamente dinâmico, com níveis que podem interagir em diversos momentos (LIDDY, 2003).

Por este motivo, apresenta-se uma descrição resumida dos principais níveis de análise da linguagem escrita.

- Morfológico

Este nível analisa a formação das palavras, buscando a sua decomposição em morfemas – a menor unidade com significado. Por exemplo, a palavra predestinado pode ser morfologicamente separada em três morfemas: o prefixo *pre*, o radical *destino* e o sufixo *ado*. Uma das estratégias que o ser humano pode utilizar para descobrir o significado de uma palavra desconhecida é decompô-la em morfemas e, sendo estes conhecidos, concluir o significado da palavra. Esta estratégia pode ser utilizada também pelos sistemas de PLN, por exemplo ao verificar as desinências verbais e assim concluir o tempo em que ocorre a ação expressa pelo verbo.

- Léxico

Neste nível, o significado é interpretado pela análise individual das palavras. Existem várias abordagens para se chegar a compreensão do significado no nível de palavras, sendo a técnica de associação de etiquetas de *partes do discurso* [17] às palavras a abordagem mais comumente utilizada. Outra técnica utilizada no nível lexical é substituir as palavras que tem somente uma interpretação possível por uma representação semântica do seu significado. A natureza da representação depende da teoria semântica adotada no sistema de PLN.

O nível lexical pode requerer o uso de um léxico para a interpretação do significado. A abordagem adotada na implementação do sistema de PLN determina se um léxico será ou não utilizado, bem como o tipo e a extensão das informações nele representadas. Os léxicos podem ser bem simples, contendo somente a palavra e as suas possíveis etiquetas *part-of-speech* (*POS)*, mas podem ser também extremamente complexos e conter informações sobre a classe semântica da palavra, argumentos e respectivas limitações semânticas e até mesmo o campo semântico de cada sentido de uma palavra polissêmica.

- Sintático

Este nível analisa as palavras de uma sentença visando descobrir a estrutura gramatical da frase. Este tipo de análise requer uma gramática e um analisador. A saída produzida por este nível de processamento é uma representação que revela os relacionamentos de dependência estrutural entre as palavras de uma sentença. Estão disponíveis uma grande variedade de gramáticas, as quais determinam a escolha do analisador. A sintaxe é importante para o significado porque a ordem e a dependência entre as palavras influenciam o significado da sentença. Por exemplo: as frases “O menino feriu o cachorro” e “O cachorro feriu o menino” têm diferenças somente à nível de sintaxe, mas seus significados são muito diferentes.

- Semântico

---

[17] *Partes do Discurso* é um termo utilizado na Linguística que se refere às funções gramaticais que as palavras cumprem na sentença. Comumente referenciado na literatura técnica pela expressão em inglês *part-of-speech (POS)*.

O processamento semântico determina os possíveis significados de uma sentença pela análise das interações entre os significados das palavras que a compõe. Este nível do processamento pode incluir a desambiguação semântica de palavras com múltiplos significados, selecionando somente um sentido da palavra polissêmica para a representação semântica da sentença. Quando a informação que determina o significado de uma palavra origina-se dos demais componentes da sentença diz-se que a desambiguação ocorreu a nível semântico. Existem vários métodos para se conseguir a desambiguação em nível semântico, como por exemplo a utilização da frequência em que ocorre determinado significado num corpus específico, métodos que consideram o contexto local e aqueles que utilizam o conhecimento pragmático do domínio do documento.

- Pragmático

Este nível se preocupa com o uso intencional da linguagem e utiliza o contexto em detrimento do conteúdo textual para alcançar o entendimento. Visa explicar como um significado implícito é identificado em textos onde este sentido não é referenciado explicitamente. Requer conhecimento profundo do mundo real, incluindo a compreensão de intenção, planejamento e objetivos. O alcance deste nível de análise para aplicações de PLN pode demandar o uso de bases de conhecimento e módulos de inferência. Um exemplo clássico de uso seria a aplicação da análise pragmática para a resolução da dubiedade em referências anafóricas ambíguas resolvíveis.

2.2.2 Categorias e Aplicações da PLN

Existem quatro categorias principais de classificação para sistemas de PLN: simbólica, estatística, conexionista e híbrida (RILOFF; SCHELER, 1996). As abordagens simbólica e estatística são utilizadas desde os primórdios da PLN.

Embora a abordagem simbólica tenha predominado inicialmente, com o advento da grande disponibilização de recursos computacionais ocorrida na década de 80, a abordagem estatística ganhou popularidade, principalmente devido à necessidade de desempenho para lidar com os vastos bancos de dados do mundo real. Também nesta época, a abordagem conexionista reconquista seu espaço, demonstrando a utilidade da aplicação de redes neurais no PLN.

Qualquer aplicação que manipule textos pode fazer uso das teorias e implementações que o PLN provê para melhor alcançar seus objetivos. No campo da Recuperação da Informação, por exemplo, a intensa manipulação de documentos a caracteriza como uma forte candidata ao uso das técnicas de PLN. No entanto, o uso de PLN em sistemas comerciais via de regra restringem-se ao nível morfológico, o que limita consideravelmente a *Precisão* destes sistemas.

Uma área de aplicação mais recente da PLN é a Extração da Informação, que visa a identificação, demarcação e extração de certos elementos chaves de informação contidos em textos descritos em linguagem natural, visando a representação estruturada destas informações. Estas extrações podem ser utilizadas por aplicações tais como sistemas de visualização, mineração de dados e recuperação da informação.

Embora o PLN seja uma área de pesquisa relativamente recente se comparada a outras abordagens da tecnologia da informação, relatos de sucesso do uso desta técnica sugerem que a tecnologia de acesso à informação baseada em PLN permanece sendo uma área interessante de pesquisa e desenvolvimento de sistemas de informação.

2.3 Extração da Informação

A Extração da Informação é o campo de pesquisa que busca "o acesso inteligente ao conteúdo de documentos pela extração automatizada de informações relevantes a uma dada tarefa" (NÉDELLEC; NAZARENKO; BOSSY, 2009). Neste contexto, "tarefa" está relacionada ao domínio de interesse do sistema.

Assim contextualizado, as informações relevantes para a EI passam a ser aquelas que melhor caracterizam léxica e semanticamente os conhecimentos de um domínio específico. Em geral a EI visa a recuperação automática de informações específicas a partir de textos em linguagem natural, buscando identificar a ocorrência de uma classe particular de objetos ou eventos e, mais recentemente, buscando estabelecer os seus respectivos relacionamentos (RUSSEL; NORVIG, 2003).

Os sistemas de EI posicionam-se funcionalmente entre os sistemas de RI, que meramente localizam documentos relacionados aos requisitos do usuário, e sistemas de compreensão de textos (algumas vezes chamados de *parsers* linguísticos), os quais pretendem extrair conteúdos semânticos a partir da análise do texto (WIMALASURIYA; DOU, 2010).

De forma geral, o processo de EI contém uma fase de pré-processamento seguida por outra fase de aplicação de regras para identificar e interpretar textos em documentos. Em geral, as regras se utilizam *expressões regulares* ou algum outro tipo de padrão para a localização dos elementos textuais de interesse.

As regras normalmente são expressas declarativamente, mas podem ser elaboradas de diversas formas (NÉDELLEC; NAZARENKO; BOSSY, 2009). As mais simples buscam identificar valores individuais enquanto as mais complexas identificam diversos valores de alguma forma relacionados entre si. A busca de diversos valores relacionados demanda um formalismo relacional que permita a elaboração de regras de complexidade mais elaborada (CHAKRABARTI, 2002).

Uma das dificuldades enfrentadas pelos sistemas de EI é a elaboração de regras de extração que identifiquem todas as informações relevantes de um documento. A origem desta dificuldade está associada à necessidade de tratamento da diversidade de representação do significado contida nos textos em linguagem natural. A interpretação de textos em linguagem natural demanda o domínio de conceitos que estão muito além da reduzida capacidade dos sistemas de raciocínio computacionais atuais.

Uma possível solução para este desafio, citada por Nédellec, Nazarenko e Bossy (2009), é buscar-se a localização de alguns indícios mais facilmente identificáveis, como por exemplo a busca de padrões de caracteres pelo uso de *expressões regulares*. Esta abordagem esbarra em outras características inerentes da linguagem natural: os diferentes significados que uma palavra ou expressão podem ter (polissemia) e as incontáveis formas de expressar a mesma informação utilizando diferentes construções frasais (paráfrase).

Outra abordagem possível é submeter-se previamente os textos à análises linguísticas mais profundas, que identifiquem traços morfológicos, sintáticos e até mesmo semânticos das palavras, possibilitando assim que os sistemas de EI possam basear suas regras sobre informações abstratas e semanticamente mais significativas, obtendo-se assim melhor desempenho na identificação das informações. Segundo Nédellec, Nazarenko e Bossy (2009),

o uso de informações com um nível maior de abstração na formulação das regras também permitiria que a sua elaboração fosse mais facilmente interpretáveis por seres humanos.

No entanto, a análise de textos em linguagem natural necessita não somente das informações linguísticas do texto, mas também do contexto em que o documento foi escrito. *Contexto* pode ser tomado aqui como o *domínio de conhecimento* a que o texto se refere. A definição do contexto do documento seria possível pelo uso de ontologias que representem os diferentes significados que as palavras podem ter em cada domínio de conhecimento. O processo de EI poderia então beneficiar-se do uso de ontologias, buscando nelas os aspectos contextuais das palavras, como é demonstrado na próxima seção.

2.3.1 Extração da Informação Baseada em Ontologia

No seu surgimento, a EI foi definida como um processo de extração de tipos bem definidos de informações específicas a partir de um conjunto de textos restritos a um domínio (NÉDELLEC; NAZARENKO; BOSSY, 2009). Estas informações eram então utilizadas para o preenchimento de valores de campos pré-definidos, sendo então as implementações de EI consideradas como sistemas superficiais de compreensão de textos, restringindo-se somente a análise léxica dos termos.

Verificou-se que os sistemas de EI limitados a análise superficial de textos eram na prática simples filtros de palavras-chaves, o que restringe profundamente a sua capacidade de identificação de informações. A transposição a estas limitações demandam abordagens que se utilizem de análises de textos mais profundas e baseadas em conhecimento ontológico da área de domínio.

Normalmente, os sistemas de Extração da Informação Baseados em Ontologias (EIBO) possuem três subprocessos bem definidos (NÉDELLEC; NAZARENKO; BOSSY, 2009): (1) Análise do texto, que engloba desde a simples segmentação em sentenças e palavras até a análise linguística mais profunda, normalmente fazendo uso de técnicas de PLN; (2) Seleção de regras, que ocorre quando os sistemas de EI buscam no texto algum elemento textual que possa ser o fator indicador de uma sequência de condições a serem verificadas a fim de certificar-se que foi encontrada informação relevante; (3) Aplicação de regras, processo no qual ocorre verifica-se a presença de informação relevante e o devido registro da referência a elementos de interesse no texto.

Wimalasuriya e Dou (2010) consideram que, devido ao número expressivo de pesquisas que investigam o uso de ontologias no processo de extração, pode-se considerar que a EIBO já é uma subárea da EI. O processo de extração da EIBO visa sobretudo o mapeamento de segmentos do texto a conceitos e relacionamentos expressos na ontologia (NÉDELLEC; NAZARENKO; BOSSY, 2009). Este mapeamento pode ser formalizado diretamente, através da inserção de anotações no texto, ou indiretamente, através de referências inseridas na própria ontologia.

A anotação de texto baseia-se na inserção de marcas diretamente dentro do texto, alterando o seu conteúdo, demarcando segmentos de acordo com os objetivos do processo de extração. A forma indireta não altera o texto original porque insere uma referência ao segmento do texto na própria ontologia, criando uma instância do conceito ou relacionamento ao qual o segmento se refere.

Verifica-se que os processos de EI evoluíram da simples extração de palavras-chaves para o processamento semântico de textos, utilizando-se principalmente de técnicas de aquisição do conhecimento combinadas com ontologias e Processamento de Linguagem Natural (NÉDELLEC; NAZARENKO; BOSSY, 2009).

Observa-se assim que a busca pelo melhor desempenho para o processo de EI passa pelo uso das informações do nível léxico combinadas com os demais níveis de processamento da linguagem, buscando nas ontologias o referencial de conceitos para o processamento semântico e pragmático dos termos.

A relação com os sistemas geradores de anotação de textos (os *parsers*) é evidenciada pela dependência das informações linguísticas para o alcance de melhores desempenhos no processo de EI.

Neste trabalho, buscou-se implementar uma abordagem de EI que primasse pela independência do padrão adotado pelo parser para a geração das anotações linguísticas. Para alcançar-se esta almejada independência, utilizou-se um modelo conceitual para implementar uma camada de abstração entre o sistema de regras de extração e o padrão de anotações. Esta camada de abstração é concretizada pela adoção de um modelo conceitual de representação para as anotações linguísticas, o qual é a seguir apresentado.

2.4 O Modelo Conceitual OLiA

O modelo conceitual OLiA tem como objetivo permitir a interoperacionalização de diferentes modelos de anotação linguística. Para isto, implementa uma arquitetura modular de ontologias expressas na linguagem OWL, formalizando diversos níveis de mapeamento entre vários padrões de anotação linguística, sendo um *Modelo de Referência* a repositórios terminológicos existentes (CHIARCOS, 2012b).

A arquitetura do modelo OLiA contempla o uso de quatro tipos de ontologias, assim agrupadas: (1) o grupo Modelo de Referência, que especifica a terminologia comum entre os diferentes esquemas de anotação linguística, derivando-se dos repositórios terminológicos existentes, estendendo-os de acordo com os respectivos esquemas de anotação; (2) as ontologias do Modelo de Anotação, que formalizam os esquemas e etiquetas de anotação; (3) o Modelo de Ligação, que define relações de equivalência entre os conceitos e propriedades dos modelos de Referência e de Anotação; (4) as ontologias do grupo de Modelos de Referência Externos, que tem como função possibilitar a integração do OLiA a outros modelos de referência.

Segundo Chiarcos (2012b), o modelo OLiA inclui 32 Modelos de Anotação para cerca de 70 línguas diferentes, incluindo diversos esquemas de anotação multilíngues, tais como o EAGLES (CHIARCOS, 2008) que abrange 11 línguas da Europa Ocidental, o TEXT/East (CHIARCOS; ERJAVEC, 2011) que inclui esquemas para as 15 principais línguas do leste-europeu. Além destes, o modelo OLiA contém também esquemas de anotação linguística para o árabe, basco, chinês, estoniano, finlandês, húngaro, turco, entre outros. Além destes idiomas, abrange também línguas utilizadas na África, Américas, Pacífico e Austrália,

servindo-se para isto do Modelos de Anotação desenvolvidos para documentação tipológica e linguística.

O Modelo de Referência Externo do OLiA inclue ligações com os reconhecidos modelos referenciais de anotação linguística GOLD (CHIARCOS, 2008) e ISOcat (CHIARCOS, 2010), permitindo assim que qualquer conceito representado no modelo de referência do OLiA possa ser interpretado no contexto dos modelos GOLD e ISOcat. Isto significa que aplicações desenvolvidas com base no OLiA podem interoperar livremente com estes modelos de anotação.

No contexto da metodologia desenvolvida neste trabalho, a aplicação do Modelo Referencial do OLiA ocorre pela sua utilização como uma cada de abstração na formalização das regras linguísticas da ontologia de extração. Decorre daí que, em teoria, as regras linguísticas de extração podem ser aplicadas às línguas e modelos de referência externa suportados pelo OLiA, o que amplia a possibilidade de aplicação da metodologia de Extração da Informação proposta neste trabalho.

O uso do Modelo Referencial OLiA viabiliza-se pela adoção do modelo de dados POWLA (CHIARCOS, 2012[a]) para a representação dos documentos, um formalismo genérico para representar corpora linguístico em RDF e OWL, assunto da próxima seção.

2.5 O Modelo de Dados POWLA

Esta seção descreve o modelo de dados POWLA, um formalismo genérico para representar corpora linguístico em RDF e OWL/DL, projetado para suportar qualquer tipo de anotação orientada a texto (CHIARCOS, 2012a).

A ideia subjacente ao modelo POWLA é representar as anotações linguísticas em RDF e empregar OWL/DL para definir os tipos de dados e verificar a consistência das restrições existentes, adotando estes tipos de dados e restrições para representar, sem perdas, outros formalismos de representação em um formato de intercâmbio genérico.

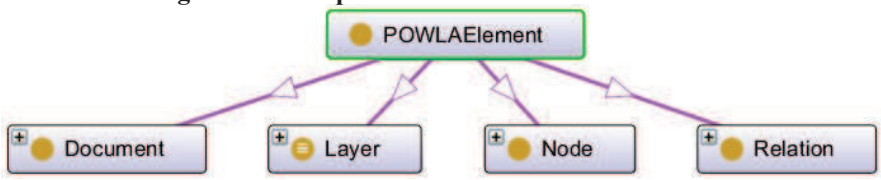
O formalismo do modelo POWLA tem dois componentes básicos: o POWLA TBox e o POWLA ABox. No POWLA TBox são representadas as estruturas de dados das anotações linguísticas, enquanto que no POWLA ABox representa-se os dados do corpus.

No modelo de dados do POWLA, formalizado no arquivo powla.owl [18], existe uma classe raiz chamada *POWLAElement* (Figura 3), ao qual todos os demais conceitos estão subordinados. As subclasses *Node* e *Relation* são utilizadas para representar as anotações linguísticas, enquanto *Document* e *Layer* estão relacionadas à organização do corpus.

A subclasse *Node* é um elemento que tem como função conter os nodos do grafo acíclico que vai representar o texto anotado na forma de uma árvore de sintaxe. As relações de hierarquia entre os nodos da árvore são definidas pelas *propriedades hasChild* e a sua inversa *hasParent*.

[18] Disponível em: <http://purl.org/powla/powla.owl>. Acesso em: 28 jul. 2013.

Figura 3: Principais conceitos do modelo POWLA.



Fonte: elaborado pelo autor.

Relation é outra subclasse do modelo relacionada à árvore hierárquica, cumprindo com a função de representar as arestas do grafo. As *propriedades hasSource* e *hasTarget* (assim como as respectivas *propriedades* inversas *isSourceOf* e *isTargetOf)* representam as relações hierárquicas entre os nodos.

Às subclasses *Node* e *Relation* podem ser associadas múltiplas etiquetas que correspondem às anotações linguísticas, o que é realizado através de *subpropriedades* derivadas da *propriedade hasAnnotation*, cujos nomes expressam o atributo da anotação. Por exemplo, a propriedade para representar a anotação part-of-speech seria uma especialização de *hasAnnotation,* tendo seu nome definido como *has_pos*.

A classe *Node* contém duas subclasses, *Terminal* e *Nonterminal*, as quais tem função de representar nodos específicos. A subclasse *Terminal* tem por objetivo representar a unidade mínima da anotação linguística sendo modelada.

Por representar as "folhas" da árvore hierárquica, a classe *Terminal* não pode estar associada à *propriedade hasChild*. A subclasse *Nonterminal* por sua vez, representa os nodos que tem pelo menos uma propriedade *hasChild.*

Ambas as classes *Terminal* e *Nonterminal* são caracterizadas pelas propriedades *hasString*, *hasStart* e *hasEnd*, estas duas últimas representando a posição do nodo em relação ao texto do corpus.

As restrições aplicáveis aos elementos do POWLA são representadas no TBox, permitindo assim realizar verificações quanto a coerência e correção da modelagem feita das anotações linguísticas. O corpus é representado no POWLA ABox como um conjunto de indivíduos que instanciam os conceitos definidos no POWLA TBox.

2.5.1 Interoperabilidade entre camadas de anotação

Os benefícios do modelo POWLA são a interoperabilidade estrutural entre as camadas de anotação em corpora multicamada, interoperabilidade entre recursos léxico semânticos e a interoperabilidade conceitual entre tipos de anotações relacionadas mas originadas de diferentes esquemas.

Estes benefícios se refletem, no contexto da metodologia sugerida nesta dissertação, na possibilidade de formalizar regras de extração baseadas em múltiplas camadas de anotação linguística. Um corpus que tenha sido anotado em relação à sintaxe e a correferência, por exemplo, permitiria que fossem elaboradas regras com maior capacidade extração do que aquelas baseadas somente nas anotações de sintaxe.

Outra possibilidade, consequência direta da representação em RDF, é a possibilidade de armazenar as regras linguísticas fora da ontologia de extração, formalizando-as diretamente em consultas SPARQL, por exemplo.

Muito embora a conversão das regras de extração em DL ou SWRL para SPARQL seja um campo de pesquisa a ser explorado, prevê-se que a possibilidade de elaboração das consultas diretamente em SPARQL teria como principal vantagem um melhor desempenho na aplicação da metodologia aqui desenvolvida na extração de informação de repositórios de textos muito extensos.

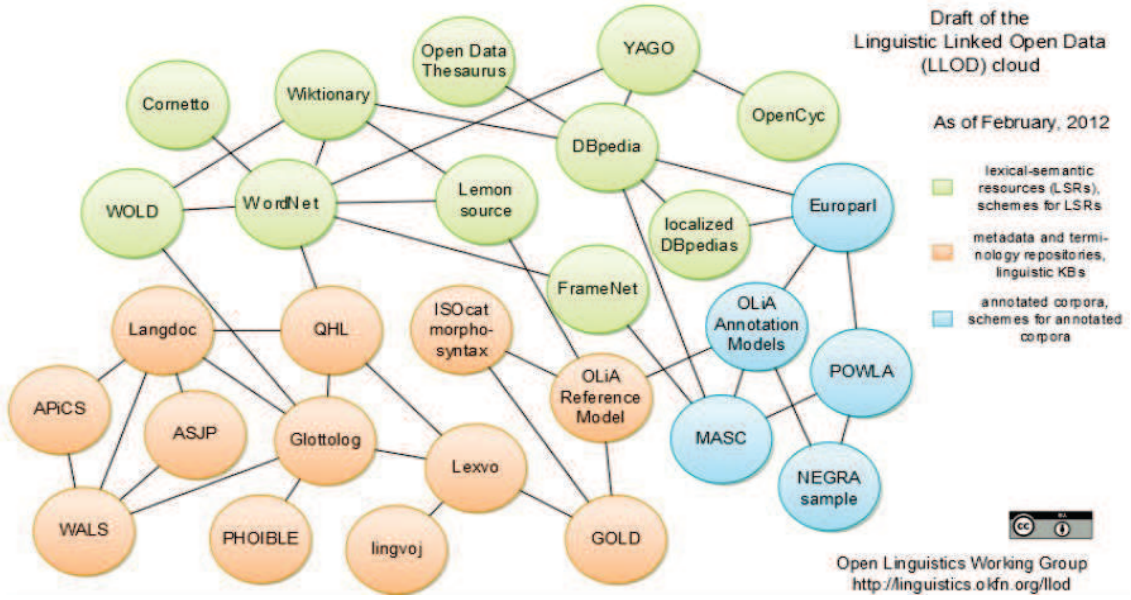
2.5.2 Interoperabilidade com Recursos Léxico Semânticos Externos

Com a possibilidade de representação de anotações multicamadas, seria possível representar-se no modelo POWLA os corpus anotados com base em recursos externos, como por exemplo a FrameNet e a WordNet, usando a representação RDF destes recursos.

Esta característica do modelo POWLA permitiria a elaboração de regras que fariam uso destas anotações semânticas para a extração de informações através de URIs (BERNERS-LEE; FIELDING; MASINTER, 1998) para estes recursos linguísticos e semânticos abertamente disponíveis na Internet. A Figura 4 apresenta uma visão geral das possibilidades de integração do modelo POWLA em relação aos recursos linguísticos abertos que compõem a nuvem LLOD[19], um dos projetos mantidos pelo OWLG[20].

A possibilidade de uso deste recursos linguísticos expande o poder de expressividade e a abrangência semântica das regras de extração da abordagem proposta neste trabalho, possibilitando a inferência sobre anotações com alto valor semântico agregado. Novamente, uma área instigante para futuras pesquisas com relação à abordagem aqui proposta.

Figura 4: O modelo POWLA na Linguistic Linked Open Data (LLOD).



Fonte: Chiarcos (2012a).

[19] Acrônimo para Linguistic Linked Open Data – http://linguistics.okfn.org/resources/llod (Acesso em: 25 jul. 2013).

[20] Acrônimo para Open Linguistics Working Group - http://wiki.okfn.org/Working_Groups/Linguistics (Acesso em: 25 jul. 2013).

Interoperabilidade Conceitual

Semelhantemente à interligação entre corpora anotado e recursos léxico semânticos externos, é possível também estabelecer amarrações entre as anotações em um corpus com repositórios terminológicos, estabelecendo assim a interoperabilidade conceitual, ou seja, as anotações linguísticas são baseadas em um único vocabulário de referência. Esta unificação referencial pode ser obtida pelo uso do modelo conceitual OLiA para a formalização dos esquemas de anotações e referência de conceitos baseados em repositórios abertos mantidos por comunidades, tais como GOLD (FARRAR; LANGENDOEN, 2010) e ISOcat (CHIARCOS, 2010).

Em relação ao contexto da proposta deste trabalho, a interoperabilidade conceitual do POWLA possibilita que as regras de extração referenciem fenômenos linguísticos independentemente do modelo conceitual adotado para a anotação linguística. Na prática, permitiria que as camadas de anotação fossem formalizadas seguindo quaisquer dos esquemas e modelos conceituais abertos. Esta liberdade em relação ao modelo conceitual implica na liberdade de escolha do parser para anotação automática de corpus.

Esta interoperabilidade conceitual possibilitaria inclusive que o corpus fosse anotado por diversos parsers linguísticos, permitindo que camadas diferentes de anotação sigam diferentes modelos de referência ou conceituais. Verifica-se assim que a adoção do modelo de dados POWLA traz consigo um leque extremamente amplo de aplicações, diversificando significativamente as possibilidades do processo de EI proposto neste trabalho.

3 TRABALHOS RELACIONADOS

Com vistas a identificar-se o estado da arte na área de EI, realizou-se uma investigação da produção intelectual existente através da revisão de trabalhos científicos relacionados, com especial atenção àqueles relacionados à EI jurídica. Este capítulo apresenta os trabalhos que influenciaram na seleção das técnicas adotadas no trabalho desenvolvido.

Os trabalhos apresentados neste capítulo tiveram como critério de seleção a implementação da EI com base em regras e que fundamentassem o seu funcionamento no uso de ontologias, visando averiguar como este artefato de representação do conhecimento é utilizado no processo de extração. Além destes dois requisitos básicos, também observou-se a atualidade do trabalho, dando-se maior preferência a trabalhos recentes. Atendidos estes requisitos de seleção, privilegiou-se os trabalhos relacionados à EI de documentos jurídicos, principalmente pela afinidade com os documentos utilizados no estudo de caso desenvolvido neste trabalho.

3.1 O sistema Ontea

No trabalho de Laclavik et al. (2012) é apresentado o sistema Ontea, uma plataforma para anotação semântica automática. O sistema utiliza expressões regulares para identificar objetos, suas propriedades ou sua posição no texto, aplicando exclusivamente o casamento de padrões como técnica de localização dos elementos de interesse.

O sistema, desenvolvido na linguagem Java, recebe como entrada o documento a ser analisado, nos formatos HTML, XML ou texto puro, e os padrões de busca, retornando o resultado em pares de atributos tipo-indivíduo e indivíduo-propriedade, os quais podem ser transformados em indivíduos OWL ou triplas RDF, através das interfaces Jena [21] e Sesame [22], respectivamente. Os conceitos identificados pelo sistema Ontea, representados na formalização apropriada, podem ser utilizados na anotação semiautomática de textos, na identificação de instâncias de conceitos ontológicos em bases de conhecimento ou para a população automática de ontologias a partir de textos.

A metodologia desenvolvida por Laclavik et al. (2012) visa trabalhar com textos específicos de uma área de aplicação e que esteja descrita por uma ontologia de domínio. Utiliza expressões regulares para a identificação de relações entre o texto e a modelagem semântica do domínio.

A avaliação do desempenho da abordagem foi realizada pela aplicação do algoritmo sobre um corpus composto por 500 documentos HTML contendo ofertas de emprego, obtidos da internet via um wrapper desenvolvido pelos próprios autores. Uma ontologia de aplicação foi criada a partir da análise manual do corpus. O corpus foi submetido ao sistema Ontea para a população da ontologia de aplicação e as instâncias criadas na ontologia foram manualmente avaliadas quanto a correção das referências.

Foram realizados três experimentos para verificação do desempenho do sistema Ontea em aplicações específicas: (1) busca de conceitos relevantes na base de conhecimento local de acordo com padrões genéricos, denominado pelos autores como experimento "Ontea"; este

[21] http://jena.apache.org/ (Acesso em: 28 jul. 2013).

[22] http://www.openrdf.org/ (Acesso em: 28 jul. 2013).

experimento obteve precisão de 64% e revocação de 83%; (2) o outro experimento de avaliação do desempenho do sistema, chamado de "Ontea creation", referia-se a criação de novos indivíduos na ontologia a partir dos elementos identificados no texto também pela aplicação pura de expressões regulares; o desempenho do sistema neste tipo de aplicação obteve índices de 28% e 81% para precisão e revocação, respectivamente; por fim, (3) o experimento "Ontea creation IR", que como o anterior cria novos indivíduos na ontologia, mas desta vez combina o casamento de padrões com cálculos de relevância das palavras para a identificação dos conceitos no texto. Neste último experimento, foram obtidos desempenho de 53% para precisão e 79% para a revocação.

Os autores consideram os resultados bastante satisfatórios, ressaltando principalmente o desempenho alcançado na identificação de instâncias na base de conhecimento local (experimento Ontea). Argumentam ainda que, embora o desempenho da metodologia adotada no sistema Ontea tenha uma alta dependência da definição das expressões regulares, o excelente desempenho em termos de velocidade de processamento da abordagem torna possível o uso da metodologia em aplicações comerciais do mundo real. Por fim, acrescentam a intenção de adicionar o uso de listas gazetteer [23] à metodologia visando incremento do desempenho geral da abordagem.

3.2 Verificação de Conformidade às Normas da Construção Civil

No trabalho de Zhang e El-Gohary (2012) apresenta-se uma abordagem para EI automática a partir de normas da construção civil para suportar a verificação automatizada da conformidade de construções. A abordagem analisa as características sintáticas e semânticas do texto através do uso de diversas técnicas de PLN, tais como listas gazetteer, tokenização, segmentação textual, análise morfológica, etiquetação POS, análise estrutural de sentenças e sintagmas.

Uma ontologia foi implementada para auxiliar a extração das características semânticas do texto. A complexidade estrutural característica das sentenças em linguagem natural é tratada com o auxílio de gramáticas livres de contexto (CFG – *Context Free Grammar*), as quais proporcionam, na opinião dos autores, a diminuição do número de padrões de sentenças necessários para a extração da informação.

A identificação dos elementos de interesse ocorre pela aplicação de regras de extração combinadas com regras para resolução de conflitos. O Capítulo 12 do Código Internacional de Prevenção de Incêndios (ICC 2009) foi utilizado para os testes preliminares da abordagem. O experimento conduzido focou-se na extração dos requisitos quantitativos de conformidade definidos no corpus.

Os autores pretendem, neste trabalho, explorar a efetividade da utilização das características sintático-gramaticais e semânticas do texto, utilizando ferramentas e técnicas de PLN para extrair informações automaticamente de códigos e normas da construção civil.

A metodologia utilizada para a condução do experimento compõe-se de um processo com 6 fases: pré-processamento, caracterização do domínio, análise da informação de

[23] Uma *lista gazetteer* consiste de uma sequência de nomes de entidades, tais como cidades, empresas, dias da semana, etc. Esta lista é utilizada para localizar ocorrências das entidades em textos (conceito disponível em: <http://gate.ac.uk/userguide/sec:gazetteers:intro>. Acesso em: 25 Jul. 2013.).

interesse, desenvolvimento das regras de extração, aplicação do processo de extração e avaliação dos resultados, sendo estas últimas três fases interativas.

A abordagem sugerida em Zhang e El-Gohary (2012) propõe o uso de casamento de padrões de características semântica e sintática do texto para a extração sequencial de informações, sendo a caracterização semântica do texto baseada em ontologias de domínio. Segundo os autores, o uso da gramática CFG permite representar de forma mais expressiva as características sintáticas das frases, reduzindo assim o esforço na elaboração dos padrões de extração das regras de EI.

As regras de extração, formalizadas em JAPE[24], são baseadas no casamento de padrões sintáticos e gramaticais do texto para a identificação da informação procurada. Visam a localização de quatro tipos de informação: tópico, atributo de verificação de conformidade, relação de comparação e quantidade.

Foi utilizado o Capítulo 12 do ICC 2006 para o desenvolvimento das regras de extração e o Capítulo 12 do ICC 2009 para teste da abordagem. A plataforma GATE (CUNNINGHAM et al., 2002) foi utilizada para a implementação e teste da proposta. A ontologia utilizada no experimento foi desenvolvida pelos próprios autores, baseados na IC-PRO-Onto (EL-GOHARY; EL-DIRABY, 2010).

O experimento obteve desempenho geral de 95% de precisão e 94% de revocação. Segundo os autores, planeja-se otimizar o desempenho da abordagem pela incorporação de resolução de correferências e utilização da análise de dependência para resolver ambiguidade de termos.

3.3 Extração Semântica de Casos Jurídicos envolvendo Contratos de TI

No relato sobre o desenvolvimento de um método de representação e inferência do conhecimento semântico visto em Maarek (2010), são vistas as informações preliminares do seu trabalho, cujo objetivo é a extração de informações semânticas a partir de um conjunto de resumos de casos jurídicos franceses ocorridos entre os anos de 2000 e 2009 (HARDOUIN, 2009), os quais relatam casos jurídicos envolvendo questões de contratos relacionados à Tecnologia da Informação (TI). Cada resumo destaca a decisão legal e os fatos levados em consideração para a tomada de decisão.

Os resumos foram realizados por advogados e compõem-se de meta informações sobre os casos, tais como: data da decisão, a decisão apelada, as partes envolvidas no caso, o embasamento legal, o entendimento (decisão final positiva ou negativa), a situação fática (a situação concreta que deu origem à questão judicial) e os comentários publicados na mídia especializada a respeito do caso [25]. O trabalho de Maarek (2010) foca-se na interpretação semântica da situação fática e do entendimento, os quais contém descrições em linguagem natural relacionados a cada caso.

---

[24] http://gate.ac.uk/sale/tao/splitch8.html#chap:jape (Acesso em: 30 out. 2013).

[25] *Embasamento legal, situação fática* e *entendimento* são termos do domínio jurídico brasileiro e foram utilizados como tradução livre para as expressões em inglês *foundation, contribution* e *solution* utilizadas no trabalho de Mareek (2010), tomando como base as informações que constam no Manual de Indexação de Jurisprudência da Justiça Federal (GUIMARÃES; BASÍLIO; DE SORDI, 1996).

Maarek (2010) propõe em seu trabalho uma representação genérica para o entendimento e a situação fática a fim de conseguir o agrupamento e localização de casos semelhantes quanto aos embasamentos jurídicos alegados pelas partes. Compõem a representação proposta a descrição do contrato, a obrigação e os fatos pertinentes apresentados à corte.

O entendimento é representado pela relação entre a decisão e respectiva situação fática pertinente. O autor propõe quatro construtos para a representação do entendimento: (1) *Ataque*: para convencer o juiz que a outra parte não cumpriu com sua obrigação *O*, faz uso do fato *F* e ganha o caso; (2) *Defesa*: para convencer o julgador de que a outra parte não cumpriu com sua obrigação *O*, faz uso da situação fática *F* mas perde o caso; (3) *Consolidação*: para convencer o juiz do cumprimento da obrigação *O*, a parte faz uso do fato *F* e ganha o caso; e, por fim, (3) *Coibição*: para convencer o juiz de que houve o cumprimento da obrigação *O*, a parte faz uso da situação fática *F*, mas perde o caso.

As decisões tomadas em cada um dos casos relatados no resumo são então associadas a um dos construtos de entendimento. Maarek assume que os advogados, ao resumirem o caso, tenham identificado os elementos pertinentes do litígio e identificado o grau de novidade e significância de cada caso.

O método de extração proposto em Maarek (2010) compõe-se de três passos: (1) sequenciamento do texto; (2) identificação das sequências e (3) reconhecimento das estruturas narrativas. O resultado da extração é validado pela verificação da coerência entre o entendimento e as situações fáticas.

O autor ressalta que o processo de extração ainda não faz uso de técnicas de PLN, sendo então necessário que a sequência de palavras a serem identificadas tenham que ser suficientemente significativas para a interpretação direta da sua semântica. Em outras palavras, presume-se o significado das palavras unicamente da sua composição léxica.

Os conjuntos de sequência de palavras e esquemas de narração foram construídos incrementalmente, sendo identificadas para cada resumo as subsequências de palavras mais importantes, nomeadas e categorizadas de acordo com o seu papel no texto. A partir da identificação destes conjuntos, definem-se as regras de identificação da estrutura da narrativa do resumo. O conjunto de sequências e regras são então utilizados pelo sistema de análise de texto para extrair o entendimento e os fatos do sumário.

Existem cinco categorias de sequências de palavras: (1) Natureza do contrato; (2) Obrigações decorrentes do contrato, da lei ou de princípios; (3) Fatos utilizados para embasar a decisão; (4) os Elementos que compõem a argumentação; e, por fim, (5) os Modificadores. As três primeiras categorias compõem-se dos elementos dos resumos, sendo as duas últimas categorias compostas pelos participantes da narração.

No trabalho analisado, Maarek ressalta que não fazia uso de uma ontologia para a representação das categorias, o que leva a concluir que a adoção da representação da taxonomia das categorias sob a forma ontológica é uma tendência no decorrer do amadurecimento da abordagem proposta. Esta observação reforça a opção pela utilização de ontologias para a representação do conhecimento em sistemas de EI.

Além disto, o autor também frisa que o uso de técnicas de PLN poderia melhorar o desempenho do processo de extração e estender o grau de reutilização e generalização das

listas de sequências de palavras utilizadas na metodologia sugerida, explicitando os benefícios que o PLN agrega ao processo de EI.

3.4 Extração de Elementos Legais a partir de Textos Jurídicos

O trabalho de Wyner (2010) apresenta um experimento para o desenvolvimento e aplicação das ferramentas de PLN a casos jurídicos para produzir textos semanticamente anotado com vistas a facilitar o processo de Extração da Informação, focando-se nos elementos constituintes do caso legal em detrimento às situações fáticas do caso.

O estudo de caso conduzido em Wyner (2010) baseou-se nas análises que Bransford-Koons (2005) realizou sobre 47 casos criminais retirados da Suprema Côrte da Califórnia e do Tribunal de Apelações daquele estado. O estudo de viabilidade realizado por Wyner baseou-se na análise de dois casos jurídicos reais. Foram utilizados os módulos Tokeniser, Gazetteer e Java Annotation Patterns Engine (JAPE) da plataforma de processamento de linguagem natural GATE (CUNNINGHAM et al., 2002).

A estratégia adotada em Wyner (2010) para a anotação do corpus fundamentou-se na utilização de listas manualmente elaboradas para a captura de padrões simples não sistemáticos, deixando a captura de padrões mais complexos e sistemáticos por conta das regras implementadas em JAPE.

Listas gazetteer são utilizadas para anotar trechos do texto, formando o que o autor chama de camada inferior de anotação. Sobre esta camada estão baseadas as regras JAPE de extração, denominadas de camada superior de anotação. O conteúdo das listas e regras de extração dependem do contexto do corpus, sendo necessário "personalizá-las" de acordo com a origem dos documentos analisados.

As anotações da camada inferior foram implementadas utilizando o módulo "Flexible Gazetteer" da plataforma GATE. As regras JAPE da camada superior de anotação podem simplesmente traduzir uma sequência de caracteres em uma anotação simplificada ou, considerando o contexto, marcar conceitos semânticos mais complexos. Wyner ressalta que planeja futuramente desenvolver a sua abordagem pela incorporação de uma ontologia ao processo, relacionando-a às informações extraídas pelo seu processo.

3.5 Extração de Regras a partir de Regulamentos

O trabalho apresentado em Wyner e Peters (2011) relata a continuidade do trabalho de Wyner (2010), buscando desta vez a identificação e extração de regras condicionais e deônticas a partir da análise de textos legislativos, especificando antecedentes, consequentes, agentes, temas, ações e exceções.

A abordagem apresentada em 2011 segue com as mesmas características técnicas de 2010, utilizando os mesmos módulos de PLN da plataforma GATE (CUNNINGHAM et al., 2002) usados no experimento relato em 2010, contudo demonstrando um uso mais aprofundado das informações linguísticas na elaboração das regras JAPE, viabilizado pelo uso do parser linguístico Stanford (DE MARNEFFE; MACCARTNEY; MANNING, 2006).

No trabalho de Wyner e Peters (2011) vê-se um estudo de caso baseado no texto do Título 21, parte 610, seção 40 do Código Federal de Regulamentação dos Estados Unidos, seção de Administração de Drogas e Alimentos. Observa-se uma revisão na metodologia de experimento, desta vez apresentando alguns resultados iniciais da abordagem.

Do ponto de vista metodológico, uma das novidades do trabalho apresentado pelos autores está relacionada à inserção de um novo processo na metodologia: a análise da saída do parser linguístico em busca de caracterização gramatical para a localização de cláusulas de exceção, conceitos deônticos, verbos principais, negação, sujeitos e objetos diretos, bem como das estruturas de sentenças condicionais. Esta análise é realizada manualmente e demanda, segundo os autores, um uso intensivo de conhecimento linguístico e jurídico.

Outra novidade da abordagem é o mapeamento de papéis temáticos à padrões sintáticos alternativos pelo uso das informações gramaticais geradas pelo parser linguístico. Isto permite, por exemplo, identificar quem é o agente e o paciente de uma determinada ação nas frases escritas no modo passivo ou ativo, utilizando para isto os papéis temáticos derivados da Verbnet (KIPPER et al., 2008).

Com relação ao corpus utilizado, são realizados uma série de pré-processamentos visando otimizar-se a análise do documento pelo parser linguístico. O documento foi dividido em várias subseções, sendo estas submetidas à análise do parser. A partir do retorno dado pelo parser são identificadas as questões relativas a sentenças muito longas e/ou complexas, bem como as questões relacionadas a listas de enumerações e referências, sendo então criados documentos, que os autores denominam de derivados, em que estas estruturas linguísticas são simplificadas.

Os documentos derivados são utilizados para o desenvolvimento dos módulos que compõem a solução. No estágio do trabalho relatado em Wyner e Peters (2011), os autores informam que a metodologia foi testada somente sobre os documentos derivados. O desempenho da metodologia foi medido pela análise das anotações geradas pela aplicação da metodologia, alcançando índices gerais de 98,91% para a precisão e 73,36% para a revocação.

3.6 Extração da Informação baseada em Ontologia de Extração

Labský, Svátek e Nekvasil (2008) apresentam o sistema Ex, que explora o paradigma de EI baseado em ontologias de extração (EMBLEY, 2004)(EMBLEY; TAO; LIDDLE, 2005), apresentando como diferenciais o uso de raciocínio probabilístico para extração de candidatos a atributos e instâncias e a combinação da ontologia de extração com as abordagens indutivas e baseadas em wrapper.

Os autores apresentam como principais características do sistema (1) a possibilidade de encontrar evidências para a extração baseado em probabilidade estimada combinada com outras informações quantitativas, tais como distribuição de valores, permitindo o cálculo de verosimilhança para cada candidato a atributo ou instâncias usando inferência pseudo-probabilística e (2) o esforço em combinar ontologias de extração manualmente elaboradas com outras fontes de informação, tais como formatação HTML ou dados de treinamento.

As informações de interesse são identificadas pelo seu nome combinado com o seu tipo de dado (texto, número inteiro ou fracionário), além de vários outros indícios de extração relacionados ao valor do atributo ou ao contexto em que a informação aparece. Os indícios de extração podem ser padrões de caracteres, valores mínimo e máximo para o caso de valores numéricos, axiomas expressando restrições mais complexas para o valor ou regras de resolução de correferência.

Os padrões são representados no sistema, tanto para valores quanto para o contexto, por expressões regulares aninhadas definidas como tokens (palavras), caracteres ou tags de formatação HTML.

O sistema recebe como entrada a ontologia de extração e um conjunto de documentos, realizando o processo de extração em cinco fases: (1) pré-processamento, quando podem ocorrer a análise da estrutura do documento, tokenisação, lematização, segmentação de sentenças e opcionalmente a execução de um etiquetador *POS* ou reconhecedores de entidades nomeadas externos. (2) Geração de atributos candidatos, baseado em padrões de valores ou de contexto. (3) Geração de instâncias candidatas para as classes alvo. (4) Indução de padrões de formatação, que permite explorar características locais que se repetem com regularidade, como por exemplo uma tabela com 100 linhas listando nomes dos componentes de equipe cuja primeira coluna tenha reconhecido pelo sistema nomes de pessoas em 90 linhas, é razoável então induzir-se que as 10 linhas restantes contém também nomes de pessoas, caso estes não sejam reconhecidos como tal. (5) Análise de atributos e instâncias, consistindo na pesquisa dos candidatos por via programação dinâmica. Uma lista das instâncias e atributos mais prováveis é retornada nesta fase.

As ontologias de extração são manualmente elaboradas por engenheiros do conhecimento experientes, requerendo semanas de trabalho para o projeto inicial. Entre as possibilidades investigadas para a redução da quantidade de trabalho requerido está a utilização de ontologias de domínio como ponto de partida para a construção semiautomática das ontologias de extração.

A metodologia apresentada pelos autores inicia ou pela adoção (ou construção) da ontologia de domínio ou diretamente pela elaboração da ontologia de extração. Contudo, em optando-se pela elaboração direta da ontologia de extração, torna-se necessário o desenvolvimento da ontologia de domínio a ser populada, sendo então proposto pelos autores duas possibilidades: (1) a ontologia de domínio pode ser derivada da ontologia de extração ou (2) assumir que a ontologia de extração pode ser sintaticamente transformada em uma ontologia de domínio, com a restrição de que tal ontologia é uma abordagem profundamente orientada a documentos.

O desempenho da abordagem sugerida por Labský, Svátek, Nekvasil (2008) é avaliado pela aplicação da metodologia a um corpus de 485 anúncios de seminários científicos, previamente anotados, sendo 240 disponibilizados ao projetista da ontologia de extração e 245

utilizados para a realização de testes. Apresenta-se como índice de desempenho a métrica *F-measure* para a localização de três elementos de interesse: o nome do palestrante, o local, a hora de início e fim do evento. O sistema obteve 94% para a hora de início e fim do evento, 69% para a extração do palestrante e 77% para a identificação do local do evento, desempenhos que estão, segundo os próprios autores, aquém dos observados em algoritmos de extração de documentos HTML baseados em indução.

A fim de cumprir com o objetivo de apresentar índices de desempenho comparáveis aos demais trabalhos apresentados nesta seção, buscou-se os índices de desempenho do sistema Ex em trabalho anteriormente apresentado (LABSKÝ et al., 2007). No trabalho de 2007, o sistema Ex foi aplicado a um corpus de 109 documentos HTML retirados de sites da área médica. O corpus foi manualmente anotado, identificando a presença de 6.930 entidades nomeadas de 10 tipos diferentes. A ontologia de extração foi elaborada a partir da análise de 30 documentos deste corpus, sendo estes documentos também utilizados para a elaboração de listas gazetteers contendo nomes de cidades e nomes/sobrenomes de pessoas comumente utilizados. Para este experimento, a metodologia apresentou desempenho médio de 67,09% de precisão e revocação.

3.7 Extração de Semântica a partir de Leis Retificadoras

O trabalho de Mazzei, Radicioni e Brighi (2009) propõe uma abordagem que utiliza análise sintática combinada com análise semântica superficial baseadas em uma taxonomia de leis retificadoras para incrementar documentos legais com metadados semânticos. A proposta utiliza o Turin University Parser (TUP) (LESMO, 2007) para construir árvores sintáticas das sentenças e um interpretador de regras semânticas para preencher frames que representam superficialmente a semântica destas sentenças.

Do ponto de vista teórico, o trabalho visa demonstrar que a combinação das análises sintática profunda e semântica superficial é uma abordagem adequada para a EI de documentos de domínios específicos, uma vez que a linguagem destes textos são mais controladas. Além disto, argumentam os autores, o uso do conhecimento especializado e linguístico tácito do domínio pode ser representado através de artefatos computacionais, como taxonomias ad-hoc.

O corpus utilizado no experimento compõe-se de leis que alteram outras leis, representadas no padrão NIR [26]. Este padrão define alguns elementos estruturais que são usados para marcar as principais partes do texto, bem como as suas subdivisões, tais como artigos, parágrafos, subparágrafos e itens numerados.

O sistema proposto pelos autores anota as modificações em um processo composto de três passos: (1) o primeiro passo é a coleta dos documentos, quando os arquivos XML são processados a fim de selecionar somente os trechos de texto em que seja útil aplicar o processo de EI, sendo os demais elementos do texto desconsiderados, sendo também nesta fase realizadas algumas alterações no texto a fim de otimizar-se o funcionamento do parser linguístico; (2) submissão do documento ao parser linguístico para análise; (3) sendo então finalmente realizada a interpretação semântica através de um sistema baseado em regras que

[26] http://www.normeinrete.it/

verifica a saída produzida pelo parser sintático em relação à taxonomia de leis retificadoras, preenchendo os slots dos frames semânticos a partir de padrões sintáticos encontrados na sentença. A busca do frame semântico a ser verificado se dá pelo verbo da sentença.

O desempenho do sistema foi avaliado com base no percentual de leis modificadoras corretamente identificadas em relação a dois quesitos: (1) tipo de alteração e posição no texto e (2) os quesitos da métrica 1 combinados com os trechos a serem modificados na lei alterada e o trecho que passa a ser a nova lei em vigor da lei modificadora. O sistema foi aplicado sobre um corpus de 181 arquivos, contendo 2.148 leis retificadoras no padrão NIR, manualmente anotadas por especialistas jurídicos. Obteve-se neste experimento um desempenho geral de 93,6% de precisão e 76,9% de revocação no experimento 1 e 82,2% de precisão e 67,5% revocação com relação à métrica 2.

Em nova publicação recente sobre este trabalho (LESMO et al., 2013), agora referido como sistema TULSI, relata-se um novo experimento, sobre um corpus de 177 arquivos, com desempenhos de 98,35% de precisão e 86,22% de revocação para a métrica 1 e 93,54% de precisão e 82,00% de revocação para a métrica 2. Pode-se observar que há uma melhora geral no desempenho da abordagem, muito embora não seja objetivo do trabalho de 2013 comparar o desempenho com a versão apresentada em 2009.

3.8 Anotação de Documentos e População de Ontologias

A solução proposta no trabalho de Amardeilh, Laublet e Minel (2005) faz uso de duas aplicações comerciais: a plataforma para gerência de conhecimento chamada Intelligent Topic Manager[27] (ITM) e o analisador linguístico Insight Discoverer Extractor (IDE). Para a integração destes dois sistemas, os autores sugerem um processo baseado nos seguintes passos: 1) exame da árvore conceitual resultante da análise linguística; 2) definição de regras de aquisição de marcas linguísticas para seu mapeamento em conceitos da ontologia; 3) aplicação automatizada destas regras nos documentos.

Os autores salientam que no projeto apresentado verifica-se que as ferramentas linguísticas e a ontologia de domínio podem ser modeladas de forma totalmente independentes e, para ilustrar esta afirmação, apresentam os resultados da aplicação da solução no domínio jurídico. Embora sejam processos separados, os autores salientam que a modelagem da representação é de suma importância para que o sistema de captura de conhecimento e anotação semântica de documentos proposto alcance um desempenho satisfatório.

Observa-se neste trabalho, principalmente devido ao desempenho, mas também por questões relacionadas com reutilização, integração e interoperação semântica, o uso de uma Base de Conhecimento para o armazenamento das instâncias dos objetos da ontologia.

O posicionamento defendido pelos autores de que a ontologia e o método de extração da informação são independentes entre si é um ponto importante a ser analisado, pois reforça a ideia de que as ontologias têm como objetivo principal a representação do conhecimento de um domínio, cuidando-se para que não ocorra uma fusão da ontologia com informações específicas da aplicação. Ontologias de domínio independentes da aplicação são naturalmente reutilizáveis. Por outro lado, reforçam, ontologias que contenham dados de aplicação terão

[27] http://www.mondeca.com/Products/Intelligent-Topic-Manager (Acesso em: 30 out. 2013).

prejudicadas a sua reutilização, criando uma dependência funcional entre ontologia e aplicação.

Anotações semânticas em documentos e população de ontologias a partir de extrações linguísticas é o tema central do trabalho desenvolvido em Amardeilh, Laublet e Minel (2005), no qual é apresentada uma arquitetura para a população semiautomática de ontologias a partir de documentos. Na arquitetura proposta foi utilizado o formato padrão RDF para a descrição de recursos (anotação semântica dos documentos), OWL para modelar a ontologia e Topic Maps (PARK; HUNTING, 2003) para a implementação da Base de Conhecimento.

A Base de Conhecimento contém as instâncias dos conceitos, propriedades e relacionamentos descritos na ontologia de domínio. Todo o conhecimento pertinente ao domínio contidos nos textos são capturados para instanciação na Base de Conhecimento e para a inserção de anotações semânticas nos próprios documentos. Segundo os autores, estas anotações podem ser compartilhadas, publicadas, consultadas ou utilizadas para outras finalidades.

A proposta foi validada em um experimento sobre um corpus composto de 36 relatórios da suprema côrte de apelações francesa, sendo 4 relatórios utilizados para a definição de 72 regras de aquisição para o reconhecimento de sete classes de tópicos, 17 tipos de atributos, uma associação e dois papéis. Sobre os 32 documentos restantes aplicou-se a metodologia sugerida, obtendo uma *Precisão* de 79% e *Revocação* de 55% a partir de uma conferência realizada de forma manual.

Os autores apontam como objetivo futuro o desenvolvimento de uma linguagem formal para descrever os conceitos necessários para a população da ontologia a partir das árvores conceituais geradas pelo parser IDE. Os autores apontam como vantagem da incorporação desta linguagem a possibilidade de formalizar as regras de extração sob a forma declarativa.

3.9 Otimização da Representação de Casos Jurídicos via EI

No projeto SMILE (BRUNINGHAUS; ASHLEY, 2001), os autores utilizam como exemplo a análise de contendas na área de Direito Comercial, mais especificamente em disputas que envolvem direitos autorais e espionagem industrial. O objetivo geral é identificar elementos de informação utilizados pelo sistema CATO, um programa de computador para análise de argumentação baseado em casos.

Em um primeiro momento, utilizou-se no projeto SMILE técnicas de Aprendizagem de Máquina para a identificação dos elementos, contudo obteve-se um desempenho muito abaixo do necessário, pois as características da demanda não indicavam este tipo de abordagem para a identificação dos elementos de interesse.

A ideia central do trabalho apresentado fundamenta-se na concepção de que para alcançar o desempenho almejado na identificação dos elementos de interesse é necessário a utilização de uma técnica de extração que tenha seu funcionamento baseado na representação do conhecimento do domínio.

Em específico, busca-se neste trabalho de Bruninghaus e Ashley a identificação de alguns elementos característicos das contendas do Direito Comercial, tais como o reclamante,

o réu e o objeto de disputa, este último genericamente denominado pelos autores como produto.

A localização das partes envolvidas, ou seja, o reclamante e o réu, se dá pela aplicação de casamento de padrões utilizando-se de expressões regulares combinadas com *gramáticas livres de contexto*; os produtos, devido à maior complexidade envolvida na identificação, seriam localizados pela ferramenta de extração de padrões AutoSlog (RILOFF; PHILLIPS, 2004), que utiliza a análise linguística superficial do texto realizada pelo segmentador de sentenças Sundance (RILOFF; PHILLIPS, 2004) para a identificação dos elementos de interesse.

O processo de extração proposto em Bruninghaus e Ashley (2001) inicia pela submissão do texto ao segmentador Sundance, que realiza uma análise linguística superficial das sentenças, localizando os seus componentes *POS*: o sujeito, o verbo, o objeto, etc.

A saída do segmentador é submetida ao sistema AutoSlog, que tem como objetivo encontrar ocorrências de uma palavra em um determinado contexto linguístico. A ideia básica do funcionamento do AutoSlog é que, dado um conjunto de características de uma palavra alvo, é possível localizar outros termos que a ela se referem ou estão relacionados.

Apresenta-se no trabalho de Bruninghaus e Ashley (2001) somente o desempenho da abordagem na identificação automática do produto, restringindo-se os autores a apenas detalhar a metodologia adotada na identificação das partes (reclamante e réu).

O experimento de avaliação apresentado compõe-se de um processo em duas fases. Na primeira fase ocorre a elaboração manual de uma lista de produtos a partir de um corpus de treinamento. Após, o corpus de treinamento e a lista de produtos são submetidos ao sistema AutoSlog para a geração do contexto linguístico das sentenças que contém referências aos produtos.

Tendo as listas de produtos e respectivos contextos linguísticos, submete-se ao AutoSlog as duas listas e o corpus de teste, desta vez para a realização do processo de extração, para o qual relata-se o desempenho de 66,15% para a *Precisão* e 64,82% para a *Revocação*. Não é explicitado pelos autores o número de documentos utilizados neste experimento.

3.10 Comparações e análises

Conforme citado na introdução deste capítulo, procurou-se apresentar ao leitor um panorama hodierna das técnicas utilizadas para a EI a partir de documentos em linguagem natural. Nesta seção busca-se fornecer uma visão geral das principais características dos trabalhos vistos e posteriormente traçar um paralelo entre as abordagens apresentadas neste capítulo e a desenvolvida neste trabalho.

A Tabela 3 apresenta sinteticamente as características principais dos trabalhos apresentados no decorrer deste capítulo, agrupadas em quatro categorias: propósito do trabalho, método de identificação, recursos utilizados para a extração e a base da implementação.

Tabela 3: Resumo geral dos trabalhos analisados

Principais Características	**3.1**	**3.2**	**3.3**	**3.4**	**3.5**	**3.6**	**3.7**	**3.8**	**3.9**
Propósito									
Anotação Semântica	X			X	X			X	
Popular ontologias	X					X		X	
Identificação de conceitos		X	X						
Popular frames							X		
Métodos de identificação dos elementos de interesse									
Expressões Regulares	X		X						
Padrões Sintáticos		X		X	X	X		X	X
Relevância da Palavra	X								
Gramática Livre de Contexto		X							X
Listas Gazetteer				X	X	X			X
Recursos utilizados para extração									
Ontologia de Domínio	X	X						X	
Ontologia de Extração						X			
Taxonomia ad-hoc							X		
Verbnet					X				
Implementação									
Baseada na Plataforma GATE		X		X	X				
Usa Parser linguístico					X		X	X	X

Embora a abordagem desenvolvida neste trabalho não esteja atrelada a um tipo específico de documento, para o estudo de caso optou-se pela adoção de um corpus que permitisse a análise do desempenho da abordagem aqui proposta no domínio jurídico, mas diferentemente da abordagem vista na seção 3.7, este trabalho prevê como entrada o recebimento de documentos contendo textos em linguagem natural não anotados e no formato texto puro (tal qual os apresentados nas seções 3.2, 3.3, 3.4, 3.5, 3.8 e 3.9).

A abordagem apresentada neste trabalho visa em primeira instância o mapeamento conceitual entre termos e expressões do texto e uma ontologia de domínio como visto nos trabalhos apresentados nas seções 3.1, 3.2 e 3.8; diferentemente destas abordagens, no entanto, propõe-se aqui que o processo de extração independa da terminologia utilizada na ontologia de domínio para a identificação dos elementos de interesse, assemelhando-se funcionalmente assim à abordagem proposta na seção 3.6, que prevê a implementação de uma ontologia especialmente dedicada à definição das regras de extração (ontologia de extração).

Por outro lado, operacionalmente a abordagem proposta neste trabalho fundamenta-se na análise profunda das características linguísticas do texto, neste sentido identificando-se mais com as abordagens vistas em 3.2, 3.5, 3.7, 3.8 e 3.9. Diferentemente das abordagens puramente baseadas no casamento de padrões de caracteres, como visto nas seções 3.1 e 3.3, o trabalho aqui desenvolvido busca o aperfeiçoamento do desempenho no processo de EI pela definição de regras que considerem as informações geradas automaticamente por parser linguísticos.

O trabalho aqui proposto tem como objetivo final a disponibilização de uma Base de Conhecimento (assim como os trabalhos apresentados nas seções 3.1 e 3.8), a qual conterá informações suficientes para associação dos elementos de interesse identificados a uma ontologia (mapeamento conceitual) e respectiva localização da referência ao conceito no documento original (mapeamento textual).

Assim como todas as implementações de EI apresentadas neste capítulo, a abordagem proposta neste trabalho também utiliza regras para o processo de extração. Contudo, diferentemente de todas as abordagens vistas neste capítulo, utiliza-se para formalização das regras as linguagens DL e SWRL. Desta forma representadas as regras, observa-se uma lógica unificada para a implementação do processo de extração.

Nos trabalhos analisados, uma parte da semântica das sentenças está representada na ontologia de domínio (tipicamente as relações de hiponímia, hiperonímia, holonímia, meronímia, sinonímia e antonímia entre os termos específicos do domínio), mas a desambiguação de significado ocorre em uma aplicação externa à ontologia em si. Isto significa que parte da modelagem do conhecimento está incorporada à aplicação. Como uma parte do conhecimento está representado na aplicação, ao utilizar-se as regras de EI em uma aplicação diferente, fragmenta-se o modelo, sendo necessário então reimplementar a lógica de EI na nova aplicação.

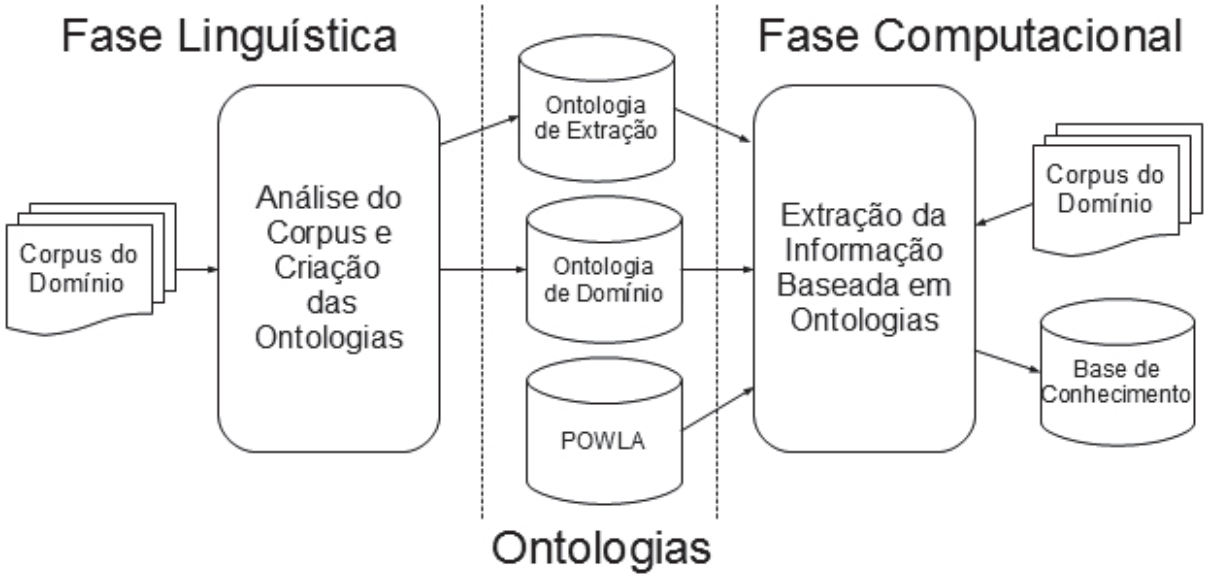
No trabalho aqui desenvolvido, a interpretação das frases em linguagem natural é realizada por consultas executadas pelo reasoner na própria ontologia de extração para a interpretação dos resultados retornados. Com esta abordagem a modelagem de todo o processo de EI torna-se realmente independente da aplicação que a utiliza.

4 MODELO PROPOSTO

Os principais componentes do modelo para Extração da Informação de documentos em linguagem natural proposto neste trabalho são apresentados nesta seção, destacando-se em especial o uso da análise linguística de corpus, a representação do conhecimento através de ontologias, a representação de corpus anotado no formato POWLA e a lógica unificada do processo de extração.

O modelo proposto, conforme ilustrado na Figura 5, subdivide-se em uma etapa linguística seguida por uma etapa computacional. A Fase Linguística fundamenta-se no estudo de corpus para criação da ontologia de domínio e das regras linguísticas de extração. A Fase Computacional faz uso de técnicas de Processamento de Linguagem Natural (PLN) combinadas com as ontologias de domínio e de extração para a implementação do processo da EI e geração da respectiva Base de Conhecimento.

Figura 5: Visão geral da arquitetura do modelo.



Fonte: elaborado pelo autor.

O conceito de ontologia de extração origina-se do trabalho de Embley (2004): artefato computacional que serve como repositório para a modelagem conceitual das regras de extração. Diferentemente da abordagem de Embley, no entanto, a metodologia aqui implementada utiliza as linguagens lógicas suportadas pelo OWL para a representação das regras de extração, possibilitando assim a sua formalização descritiva em uma lógica totalmente unificada para a implementação do processo de EI.

Além das ontologias criadas dentro do próprio modelo, nota-se a presença de um terceiro elemento ontológico: o modelo de dados POWLA (CHIARCOS, 2012a), uma ontologia que contém um formalismo baseado em OWL para representação de corpus anotado.

No contexto deste trabalho, a ontologia POWLA tem como principal objetivo a representação dos documentos do domínio no formato OWL, visando possibilitar o uso da lógica inferencial ontológica para a formalização das regras linguísticas de extração.

Na próxima seção são apresentados de forma geral os processos que compõem a Fase Linguística e, em seguida, os processos da Fase Computacional do modelo, sendo vistos em detalhes cada um dos subprocessos que o compõe.

4.1 Fase Linguística

O modelo proposto neste trabalho prevê o uso de uma ontologia de domínio e de uma ontologia de extração na Fase Computacional. Estas ontologias são criadas no decorrer da Fase Linguística.

Conforme ilustrado na Figura 6, é a partir da análise do corpus do domínio realizada pelos especialistas em linguística que são definidas as bases para a formalização da ontologia de domínio e das regras da ontologia de extração.

Figura 6: Processos da Fase Linguística do modelo.

Fonte: elaborado pelo autor.

A partir das definições formuladas pelos linguistas, o especialista em Engenharia de Conhecimento tem a tarefa de formalizar em OWL primeiramente a ontologia de domínio e posteriormente a ontologia de extração.

Nas próximas seções descrevem-se maiores detalhes dos dois principais processos da Fase Linguística do modelo: a criação da ontologia de domínio e a formulação das regras linguísticas da ontologia de extração.

4.1.1 Criação da Ontologia de Domínio

A ontologia de domínio e o seu respectivo processo de criação a partir de análise de corpus são essenciais no modelo aqui proposto, pois as reflexões elaboradas e os conhecimentos adquiridos sobre o domínio durante a fase de análise do corpus são os fundamentos sobre os quais o Engenheiro de Conhecimento baseia-se para a modelagem do conhecimento do domínio sob a forma de uma ontologia.

O processo de criação da ontologia de domínio tem início a partir da análise de corpus (1) realizada pelos linguistas, quando são identificados os elementos e relações que

Figura 7: Processo de criação da Ontologia de Domínio.

Fonte: elaborado pelo autor.

caracterizam o domínio (2), sendo estes elementos usados pelo Engenheiro do Conhecimento para a formalização da ontologia de domínio (3), tal qual demonstrado na Figura 7.

Não se define aqui o processo de criação da ontologia de domínio, embora esta seja um elemento fundamental do modelo proposto. A definição da metodologia adotada para a criação da ontologia de domínio não faz parte do escopo deste trabalho, pois, no âmbito do Projeto CNJ Acadêmico, cabe ao grupo de pesquisa em linguística a criação e formalização da ontologia de domínio, ficando então a critério deste grupo a definição da metodologia.

Aliás, o processo de EI formulado neste trabalho tem como meta buscar a independência da ontologia de domínio. Esta independência é alcançada principalmente pela associação das regras de extração diretamente aos conceitos do domínio através da ontologia de extração, não havendo relação entre o processo de EI e a terminologia utilizada para a representação ontológica do domínio, assunto este abordado na próxima subseção.

4.1.2 Formalização das Regras de Extração

Na Fase Linguística ocorre, além da estruturação da ontologia de domínio, a formalização das regras de extração, seguindo-se os procedimentos ilustrados na Figura 8. Propõe-se neste trabalho que, no decorrer da análise do corpus (1), os linguistas identifiquem nos documentos analisados as sentenças que contém elementos de interesse do domínio (2), destacando os elementos textuais e os respectivos conceitos referenciados. Com base nas sentenças do corpus devidamente identificadas pelos linguistas, o Engenheiro do Conhecimento busca estabelecer as características semânticas que darão origem às regras que deverão ser formalizadas por ele na ontologia de extração (3).

Novamente, não será aqui apresentada uma metodologia a ser seguida, pelos mesmos motivos apresentados na etapa de criação da ontologia de domínio: o detalhamento da metodologia deste processo está fora do escopo deste trabalho, pois caberá ao grupo de pesquisadores linguísticos esta tarefa.

Figura 8: Processo de criação das regras da ontologia de extração.

Fonte: elaborado pelo autor.

No entanto, baseado na experiência adquirida no decorrer do estudo de caso desenvolvido, alguns processos básicos podem ser aqui antecipados, pois a atividade de elaboração das regras demanda um conjunto mínimo de procedimentos.

Para a formalização das regras de extração, primeiramente o engenheiro de conhecimento submete as sentenças identificadas pelos linguistas a um *parser*, obtendo como retorno um texto semanticamente anotado.

A partir do texto anotado gerado pelo parser, combinado com os elementos de interesse identificados nos sintagmas pelos linguistas e a ontologia de domínio, cabe ao Engenheiro do Conhecimento identificar os padrões semânticos nas sentenças que contém referências aos elementos da ontologia para a efetiva formalização das regras de extração.

Os padrões semânticos, mencionados no parágrafo anterior, dependem do tipo de parser a ser utilizado pelo Engenheiro de Conhecimento para a caracterização das sentenças. Embora acredite-se que o retorno fornecido pelos *parsers* linguísticos profundos seja uma fonte de informação básica e essencial para a desambiguação do significado de sentenças em linguagem natural, o modelo desenvolvido neste trabalho não está atrelado a um tipo de parser em específico.

Como já citado anteriormente, a adoção do modelo de dados POWLA tem como efeito, além de diversos outros benefícios, a possibilidade de utilização de múltiplas camadas de anotação na elaboração das regras de extração.

Assim, por exemplo, é possível utilizar nas regras de extração as informações morfossintáticas geradas por um parser linguístico combinadas com anotações anafóricas geradas por um parser de correferência, sendo necessário apenas que ambas as anotações sejam representadas no formato POWLA. Isto possibilita que sejam construídas regras de extração com base em informações altamente abstratas.

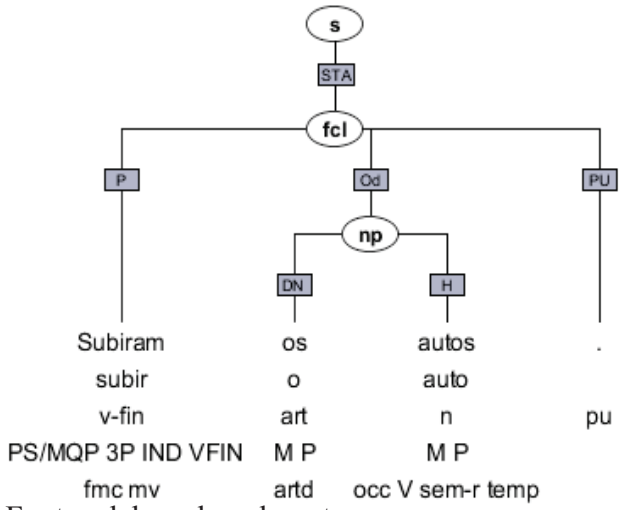
Visando uma melhor compreensão do processo, será apresentado um exemplo de criação de regras no domínio jurídico, utilizando-se as anotações geradas por um parser linguístico profundo. Será utilizada a frase "*Subiram os autos.*", a qual é extremamente característica na terminologia jurídica, comumente encontrada em documentos gerados no decorrer do trâmite processual legal. O significado desta sentença é tão específica na área jurídico que a não contextualização do domínio pode gerar interpretação de significados completamente díspares.

No contexto jurídico, "*autos*" é o conjunto de documentos que compõem um processo legal e o verbo "subir" denota que houve um deslocamento físico da documentação da instância jurídica inferior para a superior.

A frase denota então a movimentação de documentos entre as instâncias jurídicas, um evento típico que ocorre sempre que uma das partes do processo avalia que a decisão emitida por uma instância inferior não está bem fundamentada juridicamente, solicitando que a situação seja revista na instância superior.

Suponha-se que o evento denotado pela frase seja um elemento de interesse do domínio e a sua representação na ontologia de domínio da área jurídica seja expressa pelo

Figura 9: Árvore de sintaxe da frase "Subiram os autos".



Fonte: elaborado pelo autor.

conceito (classe) *Movimentação_Processual*. Neste contexto, o processo de EI deve relacionar a sentença "*Subiram os autos*." ao conceito ontológico *Movimentação_Processual*.

Seguindo-se a metodologia sugerida neste trabalho, caberia então ao linguista a marcação de frases que contenham a ocorrência do conceito e ao engenheiro de conhecimento a tarefa de buscar elementos que permitam identificar o significado da frase para então expressar estes elementos na forma de uma regra de extração.

Uma forma de caracterizar-se o significado da frase seria através do uso de informações geradas pela submissão da sentença a um *parser* linguístico. A Figura 9 ilustra graficamente as informações geradas pelo parser linguístico Palavras (BICK, 2000).

As informações linguísticas estão representadas acima por um grafo acíclico, com as etiquetas das arestas contendo as informações sintáticas (retângulos cinzas e elipses brancas), seguido abaixo pelos termos da sentença, sua forma canônica e as respectivas informações de partes do discurso[28], morfológicas e semânticas.

Com base neste conjunto de informações linguísticas abstratas o engenheiro de conhecimento poderá elaborar as regras que serão formalizadas através das linguagens lógicas do OWL e armazenadas na ontologia de extração.

Com base nas informações linguísticas apresentadas na Figura 9 é elaborado um exemplo de regra de extração, expresso em linguagem algorítmica no Quadro 1. Verifica-se que informações linguísticas do sintagma são utilizadas para a identificação do significado da frase e consequente verificação da presença ou não do conceito *Movimentação_Processual*.

No exemplo de regra apresentado no Quadro 1 estão sendo utilizadas para a verificação da ocorrência do conceito as informações sintáticas (sintagma finito e objeto

[28] A expressão "partes do discurso" é utilizada como termo técnico equivalente em português para a classificação linguística comumente referenciada pela expressão em inglês "part-of-speech" (POS).

Quadro 1: RAC linguística representada em linguagem algoritmica.

SE um sintagma finito contiver
    Um termo H cuja a forma canônica seja "auto" E
    O termo H estiver no plural E
    O termo H for objeto direto do verbo "subir"
ENTÃO
    O sintagma contém ocorrência do conceito *Movimentação_Processual.*

direto) e morfológicas (forma canônica [29], flexão de número e classe gramatical) geradas pelo parser linguístico.

O processo descrito acima é repetido para cada frase marcada pelos linguistas, resultando em um conjunto de regras de extração, as quais são formalizadas através de axiomas DL e de regras SWRL. Tais regras são armazenadas em um arquivo OWL específico, formando a chamada ontologia de extração.

A composição da ontologia de extração será vista em detalhes no estudo de caso, mais especificamente na seção 5.2. Contudo, cabe ainda aqui comentar uma característica marcante desta abordagem: a independência entre o processo de EI e a terminologia da ontologia de domínio, alcançada pela associação das regras de extração diretamente aos conceitos da ontologia de domínio.

Desta forma implementada, dispensa-se a representação na ontologia de domínio de todas as formas lexicais de um conceito. A associação semântica entre o conceito e as suas várias representações na forma escrita se dá pela elaboração de diversas regras de extração para um mesmo conceito.

Esta abordagem permite que sejam estruturadas ontologias de domínio focadas na representação conceitual do domínio, pois delega-se para a ontologia de extração a função de representar-se terminologicamente os conceitos.

Esta associação direta entre conceitos e formas terminológicas só é possível devido à formalização das regras de extração através das linguagens lógicas da ontologia, permitindo assim que as regras de extração relacionem-se diretamente aos conceitos do domínio.

Por serem as regras representadas sob a forma de axiomas e regras ontológicos e dado o fato que ontologias são reconhecidas por representarem conhecimento, referencia-se neste trabalho as regras de extração pela sigla RAC, acrônimo para Regra de Aquisição de Conhecimento.

O motivo de referenciar-se as regras de extração pelo acrônimo RAC está fundamentado no fato de que a abordagem aqui proposta permite que sejam utilizadas diversas camadas de anotação semântica para a elaboração das regras de extração.

Pode-se, por exemplo, utilizar-se um sistema de reconhecimento de entidades nomeadas (REN) combinado com um parser linguístico para a geração de duas camadas de anotação: linguística e de entidades nomeadas.

[29] A forma canônica de um termo é a sua representação sem flexão de número, gênero ou grau. Comumente chamado de forma *lematizada* do termo.

Uma vez modeladas estas anotações no formato POWLA, seria plenamente factível elaborar-se RACs que utilizassem todas as informações disponíveis para a extração de informações. Esta flexibilidade da abordagem aumenta significativamente a expressividade das RACs, aumentando o poder de reconhecimento de conceitos mais complexos.

O exemplo apresentado no Quadro 1 é intencionalmente simples, pois tem como objetivo a didática. RACs mais complexas são vistas no capítulo 5, sendo então demonstradas regras de extração que apresentam o maior poder de expressividade e flexibilidade da abordagem proposta para tratar, por exemplo, fenômenos linguísticos como a sinonímia.

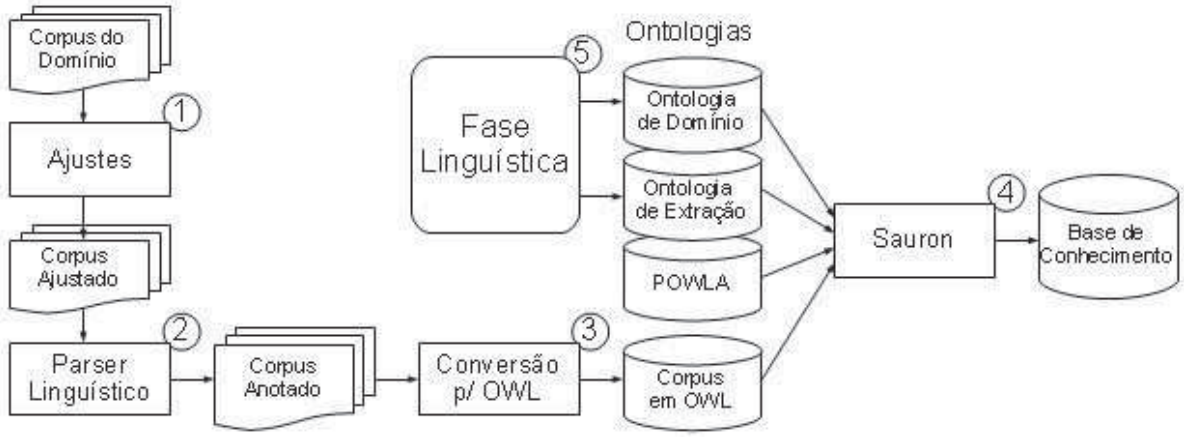
Tendo sido alcançado o objetivo de levar ao leitor uma visão geral do processo de criação das regras de extração com base em análise de corpus, dá-se início à descrição da próxima etapa, denominada de Fase Computacional, quando então são realizados os processos computacionais necessários para a efetiva extração da informação dos documentos, sendo vistos os detalhes operacionais do processo de EI baseado no uso combinado das ontologias de domínio e de extração.

4.2 Fase Computacional

A Fase Computacional do modelo, conforme ilustrado na Figura 10, compõe-se de uma sequência automatizada de processos que são aplicados ao corpus do domínio visando a realização da EI.

Primeiramente, são realizados alguns ajustes dos textos (1), sendo em seguida submetidos os documentos a um *parser* (2), o qual gera como saída o texto original acrescido de anotações abstratas, tais como anotações linguísticas, semânticas, de correferência, reconhecimento de entidades nomeadas, etc.

Figura 10: Visão geral dos processos da Fase Computacional.



Fonte: elaborado pelo autor.

Os documentos anotados são então convertidos para a linguagem OWL (3), quando então o Sistema de Aplicação Unificada de Regras e Ontologias (SAURON) dá início ao processo de EI (4), combinando as ontologias criadas na Fase Linguística (5) com o corpus representado em OWL e disparando o sistema de inferências lógicas das ontologias para

avaliação das RACs. O resultado deste processo é a geração da Base de Conhecimento com as informações sobre as referências conceituais presentes nos documentos do corpus.

Nas próximas subseções são descritas em detalhes cada uma das etapas do processo de EI da Fase Computacional do modelo proposto.

4.2.1 Ajustes dos Documentos

O primeiro processo computacional previsto no modelo aqui proposto é a realização de um pré-processamento dos documentos do corpus. Nesta etapa, conforme ilustrado na Figura 11, alguns ajustes são realizados sobre o texto, em geral visando-se ou uma melhora no desempenho dos processos subsequentes ou a correção de situações que podem comprometer a precisão da extração da informação.

Figura 11: Ajustes realizados no Pré-processamento.

Fonte: elaborado pelo autor.

Como de praxe, pequenas alterações superficiais são realizadas no texto, como por exemplo a expansão de algumas abreviaturas, correções ortográficas ou gramaticais, remoção de trechos irrelevantes ou que sabidamente causam problemas de processamento.

Neste trabalho, o principal objetivo da realização dos ajustes é a otimização da taxa de acertos do parser ao qual o texto será submetido na próxima etapa do processo. Na descrição do estudo de caso, assunto do próximo capítulo, serão explicitadas algumas alterações implementadas no decorrer do experimento.

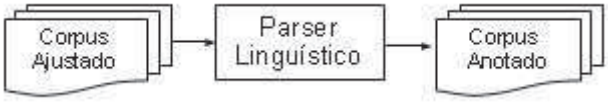
Erros, percebidos no transcorrer do desenvolvimento do estudo de caso, causados por estruturas textuais comuns e corriqueiras resultaram em problemas na fase de submissão do documento ao parser, comprometendo todo o processo de EI.

Saliente-se que são tomados os devidos cuidados para que as alterações realizadas não prejudiquem a semântica original da sentença, bem como não insiram e nem removam informações importantes do texto original. Realizados os ajustes, o documento está pronto para a próxima etapa computacional do processo: a submissão do documento ao parser para análise do texto.

4.2.2 Submissão ao Parser

Após a fase de ajustes, ocorre a etapa de submissão do documento ao parser para o acréscimo de informações através da geração de um arquivo contendo o texto original acrescido de anotações (Figura 12). Estas informações, conforme já dito anteriormente, serão utilizadas nas regras de extração para a análise do significado das sentenças no processo de Extração da Informação.

Figura 12: Processo de submissão ao Parser linguístico.



Fonte: elaborado pelo autor.

A título de exemplificação do tipo de informações que um *parser* acrescenta ao documento, são apresentadas as informações geradas pelo *parser* linguístico Palavras (BICK, 2000). Será utilizada para análise a mesma frase da seção 4.1.2. As informações geradas pelo parser para esta frase estão representadas no Quadro 2 no formato TIGER-XML (KÖNIG; LEZIUS, 2003).

O formato TIGER-XML fornece uma representação modular para papéis semânticos e estruturas morfossintáticas de frases. As frases são expressas em um grafo de sintaxe acíclico com arestas e nós etiquetados, ordenados sob a forma de uma árvore.

No Quadro 2, cada nó não terminal (*nt*) da árvore recebe uma etiqueta com uma identificação única (*id*) e a sua função sintática em relação à frase (*cat*). Para os nós terminais (*t*), os quais representam os termos que compõem a frase, temos etiquetas que mostram a sua forma canônica (*lemma*), a sua função gramatical dentro da frase (*pos*), a sua classificação morfológica (*morph*) e outras informações semânticas adicionais (*extra*)[30].

Esta etapa do processo tem como objetivo adicionar anotações ao documento original, de modo que tais informações fiquem disponíveis às regras de extração para que estas possam concluir o significado das frases analisadas.

Quadro 2: Resultado da análise da frase "Subiram os autos." em TIGER-XML.

```
1. <s id="s16" ref="16" source="Running text" forest="1" text="Subiram os autos.">
2.  <graph root="s16_500">
3.
4.  <terminals>
5.  <t id="s16_1" word="Subiram" lemma="subir" pos="v-fin" morph="PS/MQP 3P IND VFIN"/>
6.  <t id="s16_2" word="os" lemma="o" pos="art" morph="M P" extra="artd"/>
7.  <t id="s16_3" word="autos" lemma="auto" pos="n" morph="M P" extra="occ V sem-r temp"/>
8.  <t id="s16_4" word="." lemma="--" pos="pu" morph="--" extra="--"/>
9.  </terminals>
10.
11.  <nonterminals>
12.  <nt id="s16_500" cat="s">
13.  <edge label="STA" idref="s16_501"/>
14.  </nt>
15.  <nt id="s16_501" cat="fcl">
16.  <edge label="P" idref="s16_1"/>
17.  <edge label="Od" idref="s16_502"/>
18.  <edge label="PU" idref="s16_4"/>
19.  </nt>
20.  <nt id="s16_502" cat="np">
21.  <edge label="DN" idref="s16_2"/>
22.  <edge label="H" idref="s16_3"/>
23.  </nt>
24.  </nonterminals>
25.
26.  </graph>
27. </s>
```

[30]A lista completa de etiquetas está Disponível em: <http://beta.visl.sdu.dk/visl/pt/info/portsymbol.html>. Acesso em: 29 jul. 2013.

Gerado o documento anotado, tem início a próxima etapa do processo, que é a conversão do documento anotado para o formato OWL, assunto da próxima seção.

4.2.3 Conversão do Documento Anotado para OWL

O terceiro e último processo realizado sobre o documento na Fase Computacional é a conversão do texto e das anotações geradas pelo *parser* para o formato OWL (Figura 13). Esta conversão é viabilizada pelo uso do modelo de dados POWLA (CHIARCOS, 2012a) para a representação das anotações no formato OWL.

Figura 13: Processo de conversão para OWL.

O motivo de converter-se as anotações geradas pelo parser para o formato OWL fundamenta-se no fato de que as regras formalizadas na ontologia de extração podem desta forma ter acesso direto às anotações geradas pelo parser.

Esta é uma inovação metodológica proposta neste trabalho: a conversão das anotações para OWL via modelo de dados POWLA, possibilitando que as informações abstratas geradas pelo parser sejam acessadas diretamente pelo sistema de inferência da ontologia, permitindo que as regras de extração sejam formalizadas nas linguagens lógicas suportadas pelo OWL.

A conversão para OWL é realizada através do processamento automatizado do arquivo anotado e geração das triplas RDF que representam as informações abstratas geradas pelo parser.

As estruturas hierárquicas, como por exemplo uma árvore de sintaxe gerada por um parser linguístico, são representadas em definições OWL (POWLA TBox). Os dados, como por exemplo as informações morfológicas dos termos, são representados como indivíduos da ontologia (POWLA ABox).

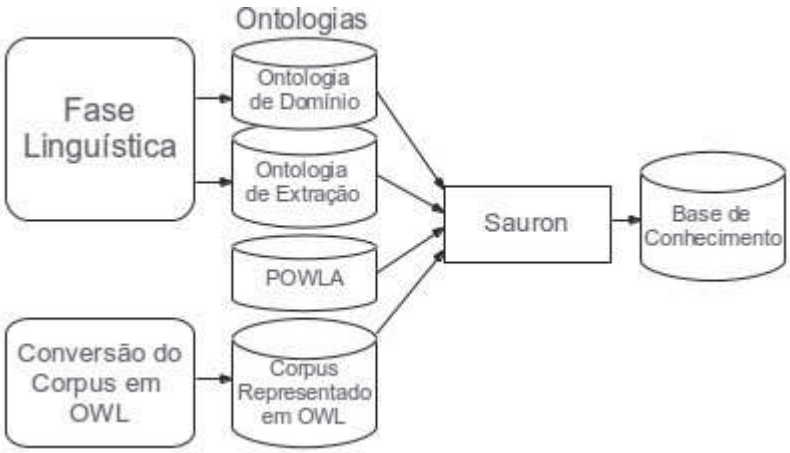
Desta forma representadas, as informações linguísticas geradas pelo parser podem ser utilizadas para a elaboração de regras expressas diretamente em DL ou SWRL na ontologia de extração, viabilizando assim a adoção de uma lógica totalmente unificada para a implementação completa do processo de extração da informação.

Na próxima seção apresenta-se o sistema que efetivamente dispara o processo de EI proposto neste trabalho.

4.2.4 Extração da Informação via sistema SAURON

O sistema SAURON é o responsável pela integração de todas as ontologias produzidas no decorrer do modelo (ontologias de domínio, de extração e o documento linguisticamente anotado representado no formato POWLA), pela execução do *reasoner* para a realização da extração de informações e pela geração da respectiva Base de Conhecimento, conforme ilustrado na Figura 14.

Figura 14: O sistema SAURON



Fonte: elaborado pelo autor.

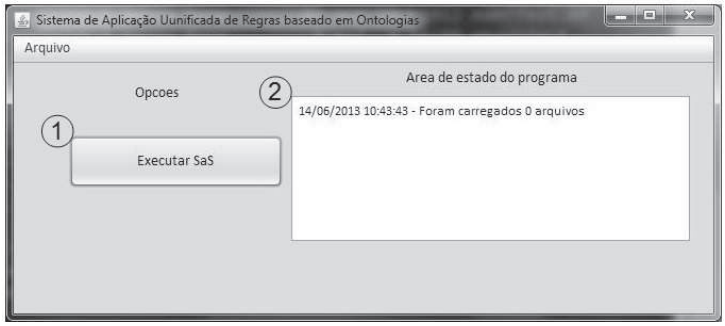
A interface do sistema é bastante simplificada, como pode ser visto na Figura 15, contendo somente um botão para a seleção dos arquivos de entrada (1) e uma área para visualização em tempo real do processamento dos documentos do domínio (2).

Uma vez que toda a lógica de extração está representada na ontologia de extração, a tarefa mais complexa executada pelo sistema SAURON refere-se a geração de uma Base de Conhecimentos que contenha somente as informações suficientes e necessárias para a localização dos conceitos do domínio no texto original.

Tanto isto é verdade que, para a elaboração e teste das regras linguísticas de extração elaboradas para o estudo de caso, utilizou-se somente o editor de ontologias Protégé para a carga/mesclagem das ontologias e posterior disparo do reasoner para a verificação da efetividade das regras.

O sistema SAURON foi implementado na linguagem Java [31] e se utiliza da interface de programação OWL API [32] para a carga/mesclagem das ontologias do modelo e posterior disparo do sistema de inferência lógica para o início do processo de EI. No Apêndice C

Figura 15: Interface gráfica do sistema SAURON.



Fonte: elaborado pelo autor.

[31] <http://www.oracle.com/technetwork/topics/newtojava/overview/index.html>. Acesso em: 23 jul. 2013.

[32] <http://owlapi.sourceforge.net>. Acesso em: 23 jul. 2013.

estão disponíveis itens mais detalhados de sua implementação, como os diagramas de sequência e atividade..

O *reasoner* processa os axiomas e regras armazenados nas ontologias, encontrando através de inferência lógica os termos e sintagmas do corpus que fazem referência a conceitos da ontologia de domínio. O processo inferencial relativo a localização das referências já foi extensivamente explicitado na seção 4.1.2.

Depois de executado o *reasoner* e inferidas as referências aos conceitos, o sistema SAURON armazena as informações na Base de Conhecimento, onde ficam armazenadas somente as informações suficientes para identificação do documento analisado, do termo ou sintagma referenciador e o conceito referenciado da ontologia de domínio.

As informações armazenadas na Base de Conhecimento são suficientes para realizar outras operações computacionais, tais como a anotação do documento original, ou a análise de contexto do documento, sendo também possível utilizar as informações da Base de Conhecimento em sistemas de RI textual para a busca de documentos.

No próximo capítulo apresenta-se um estudo de caso que demonstra a aplicação do processo de EI proposto neste trabalho sobre um corpus do mundo real, demonstrando a viabilidade e desempenho da abordagem proposta.

5 ESTUDO DE CASO E VALIDAÇÕES

No capítulo anterior foi apresentado um modelo genérico para a Extração de Informação Baseada em Ontologias. A fim de demonstrar-se a factibilidade e o desempenho do processo proposto neste trabalho, foi desenvolvido e implementado um estudo de caso em que se adotou a metodologia proposta no modelo, sendo relatados os detalhes deste experimento no transcorrer deste capítulo.

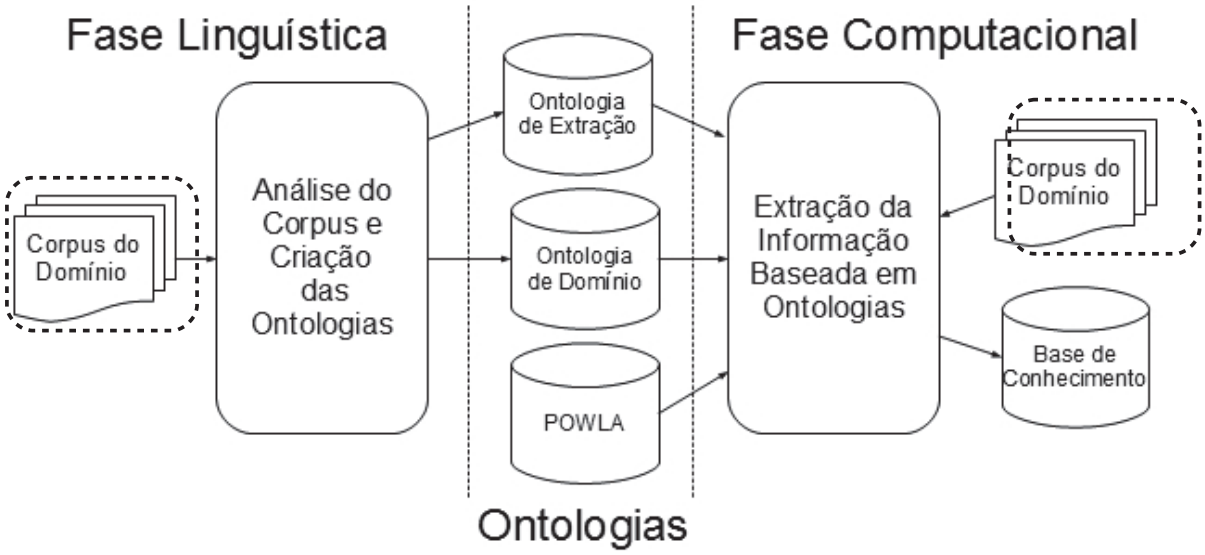
Definição do Corpus

A implementação de um estudo de caso para a avaliação do processo de EI demanda a escolha da área de domínio e de um corpus para a aplicação da metodologia, pois ambas as fases fundamentam-se nesta definição, seja para a análise do corpus na Fase Linguística, seja para a aplicação do processo de EI na Fase Computacional, como pode ser visto na Figura 16.

A escolha da área jurídica para o desenvolvimento deste experimento é uma consequência natural do contexto no qual este trabalho se encontra inserido, pois está diretamente relacionada aos objetivos do projeto CNJ Acadêmico e do grupo de pesquisa SEMANTEC.

Para a escolha do corpus a ser utilizado neste estudo de caso, no entanto, demandou-se uma análise mais ampla. Para uma melhor compreensão dos motivos considerados, faz-se necessário o conhecimento de alguns conceitos básicos do domínio jurídico brasileiro, os quais permitirão compreender a função, a origem e os detalhes do documento jurídico escolhido para a realização do experimento aqui relatado.

Figura 16: Visão geral do modelo, com destaque para o corpus..



Fonte: elaborado pelo autor.

Inicia-se então esta seção pela apresentação de uma visão geral básica do sistema judiciário brasileiro. Esta visão geral visa a contextualização e melhor compreensão do documento e de seus componentes textuais.

Por fim, apresentam-se os detalhes mais relevantes relacionados à coleta do corpus em si, visando levar ao leitor os motivos da escolha do repositório do qual foram coletados os dados utilizados para a implementação do estudo de caso aqui apresentado.

5.0.1 Uma visão geral do Sistema Judiciário Brasileiro

A composição e estrutura do Poder Judiciário no Brasil são definidos no Capítulo III do Título IV (arts. 92-126) da Constituição brasileira de 1988[33]. A Constituição também define a composição e estrutura das diversas Justiças através das quais se exercerá a função jurisdicional. Para a divisão racional do trabalho instituíram-se diferentes organismos, delegando-se a cada um deles um setor da grande "massa de causas" que precisam ser processadas (SANTOS 2002).

Essa distribuição de competência considera critérios de diversas ordens: a natureza da relação jurídica material controvertida, que determinará a atribuição dos processos à Justiça apropriada, ou a qualidade das partes envolvidas na lide (SANTOS 2002). A Figura 17 apresenta o organograma do Poder Judiciário brasileiro e os organismos que compõem a sua estrutura: a Justiça Federal, a Justiça do Trabalho, a Justiça Eleitoral, a Justiça Militar, as Justiças Estaduais ordinárias e as Justiças Militares estaduais.

Com relação aos processos que tramitam no Poder Judiciário Brasileiro, estes tem o seu fluxo regidos por meio do Código de Processo Civil e do Código de Processo Penal. A tramitação processual tem um fluxo bem definido, que estipula o rito que qualquer ação deve seguir.

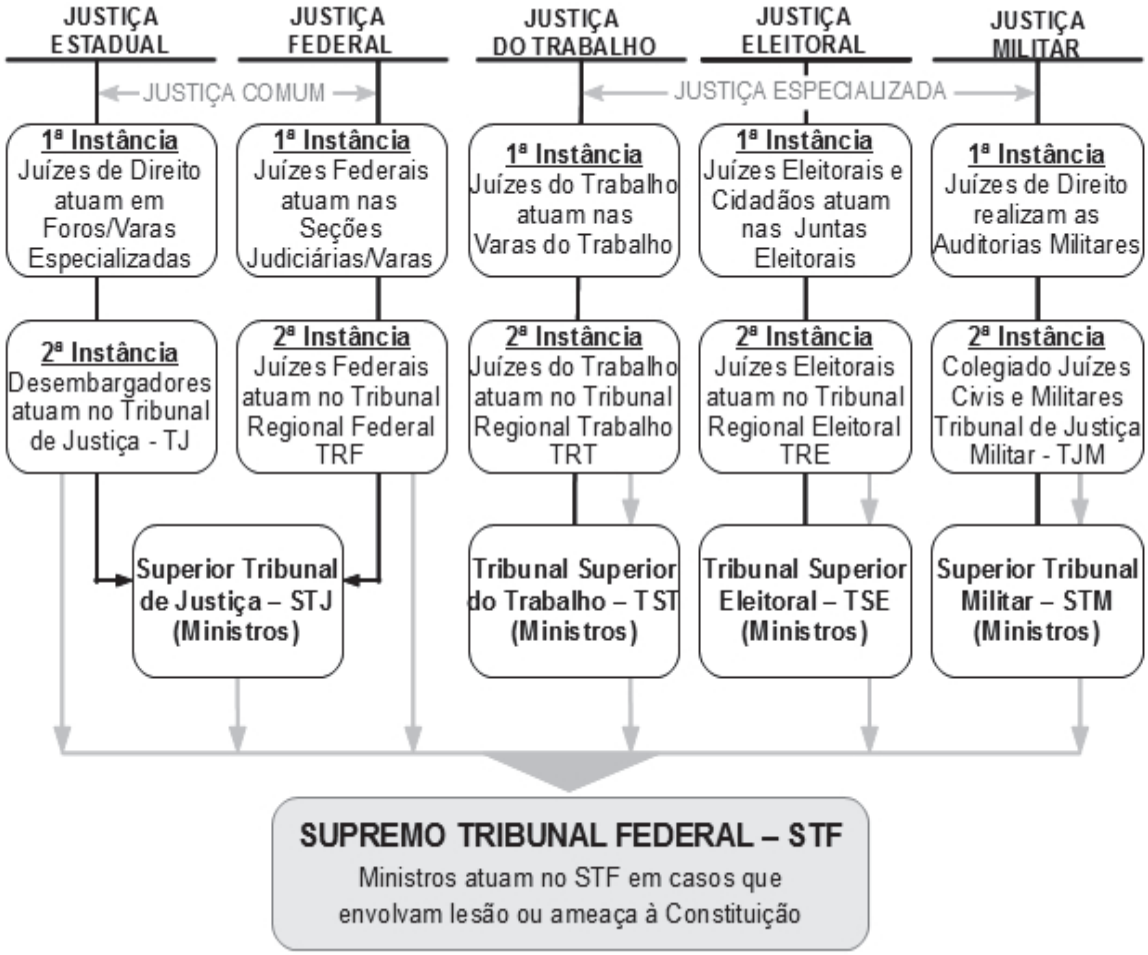
O documento utilizado no estudo de caso é aquele no qual são registradas as decisões colegiadas da 2ª Instância da Justiça Estadual. Por este motivo, será visto com maior atenção nos próximos parágrafos os detalhes da tramitação processual desta entidade judiciária, sendo ilustrado na Figura 18 o fluxo geral dos processos.

A Justiça dos Estados possui três instâncias ou graus (SANTOS, 2002), sendo a *1ª Instância* representada pelos fóruns e varas espalhados por todo o território nacional, composta por juízes de Direito que julgam os processos monocraticamente, ou seja, individualmente.

A *2ª Instância,* constituída pelos *Tribunais de Justiça* dos Estados da Federação mais o Distrito Federal, compõem de desembargadores que julgam os processos em colegiados, denominados órgãos julgadores, por meio de debate. A *3ª Instância*, denominada de *Instância Superior*, é formada pelos Ministros que compõem o Superior Tribunal de Justiça (STJ) ou Supremo Tribunal Federal (STF).

[33] Disponível em: <http://www.planalto.gov.br/ccivil_03/Constituicao/ConstituicaoCompilado.htm>. Acesso em: 10 jul. 2013.

Figura 17: Organograma do Poder Judiciário.

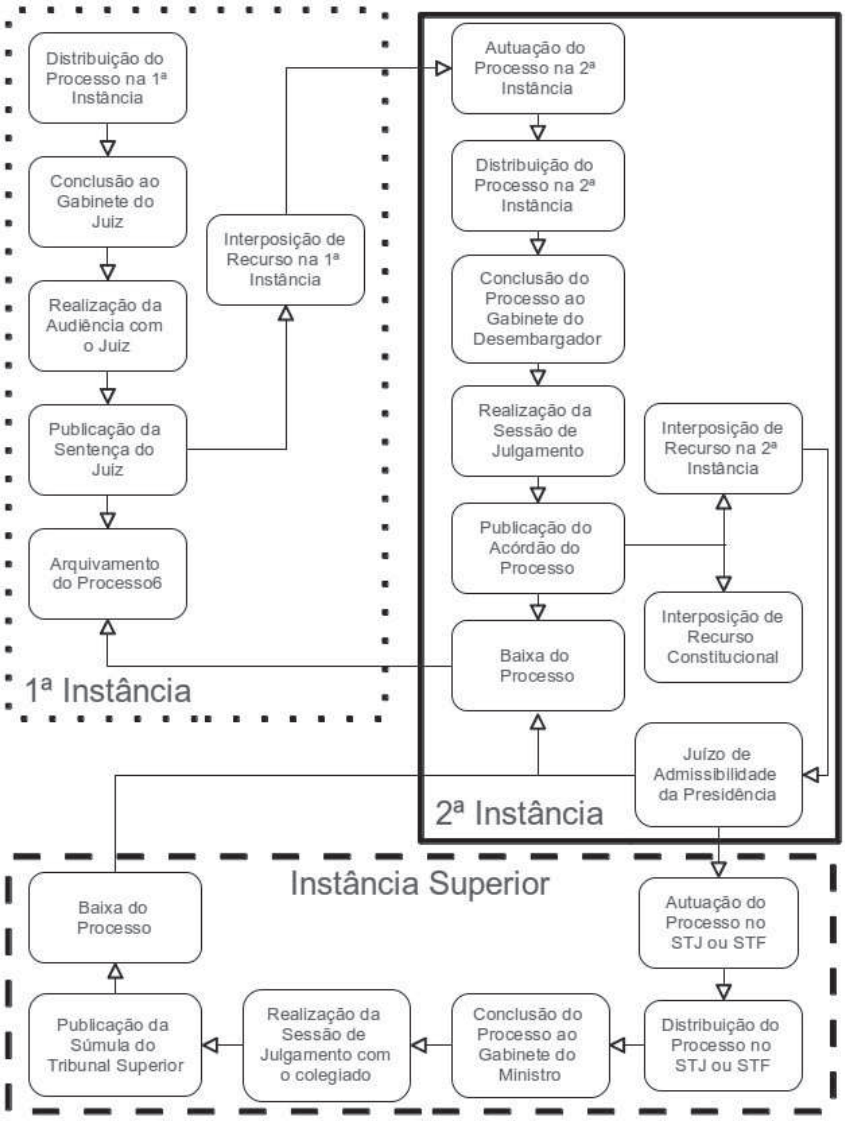


Fonte: http://direitoemamplofoco.blogspot.com.br/2012/05/organograma-do-poder-judiciario.html

Na área pontilhada da Figura 18 destaca-se o fluxo processual da *1ª Instância* da *Justiça Estadual*. Publicada a sentença do juiz de *1ª Instância*, caso ocorra discordância por qualquer das partes em relação à decisão, pode-se interpor recurso, acionando-se então a *2ª Instância* judicial, cujo o fluxo processual está representado na Figura 18 pela área com traçado contínuo.

Caso ocorra a inconformidade de alguma das partes com relação à decisão de *2ª Instância* (publicada no documento chamado *Acórdão)*, pode-se novamente entrar com recurso especial (dirigido ao STJ) ou extraordinário (dirigido ao STF), dando início ao acionamento do fluxo processual da *Instância Superior*.

Figura 18: Fluxo de tramitação processual do Justiça Estadual brasileira



Fonte: Santos (2002).

Verifica-se no fluxo processual da *Justiça Estadual* que a decisão da *2ª Instância* é publicada em um documento denominado de *Acórdão*. Conforme já comentado anteriormente nesta seção, este é o documento jurídico utilizado no estudo de caso desenvolvido nesta proposta, sendo então importante apresentar informações mais detalhadas sobre a sua estrutura, assunto da próxima subseção.

5.0.2 O Documento Acórdão

O Acórdão é um documento que contém uma macroestrutura textual definida de acordo com os arts. 165 e 458 do Código de Processo Civil [34] brasileiro, composta obrigatoriamente de um *Relatório*, da *Fundamentação* e da parte *Dispositiva,* na qual se encontra a decisão propriamente dita. Além destes elementos textuais, deverá conter também uma *Ementa,* conforme o art. 563 do Código Processo Civil, que será o *resumo* que se faz dos princípios expostos no acórdão. No Apêndice A apresenta-se, a título de exemplo, as duas primeiras páginas um Acórdão do TJRS.

Embora na prática cada gabinete de Desembargador produza um Acórdão com estilo próprio, tais documentos são em geral estruturalmente muito semelhantes nos diferentes Tribunais de Justiça do Brasil. Apenas definições de formato, sequência das sessões, fonte, entre outras superficialidades são diferenciadas.

Não há legislação para definir o formato ou conteúdo de um acórdão, permitindo que cada Tribunal, exercendo a sua independência funcional, desde que siga as definições do seu Regimento Interno, possa definir a composição deste documento. No entanto, sendo o Acórdão o documento de maior relevância para a jurisprudência, existe um maior trabalho no sentido de sua estruturação e padronização. Há um padrão tácito, na maior parte das vezes seguido pelos magistrados, possivelmente resultante de um senso comum geral. Via de regra o Acórdão é subdividido nas seguintes seções

- **Cabeçalho:** visa a apresentar metadados do processo originário. Informações tais como tipo e número do recurso, juizado de onde se originou o processo, além dos tipos (apelante ou apelado) e nomes das partes envolvidas.
- **Ementa:** é um parágrafo único constituído por um conjunto de termos, normalmente escritos em letras maiúsculas, separados por vírgulas, ponto e vírgula, ponto final, etc. Nesta seção do Acórdão costuma-se apresentar o resumo que o relator elabora após a conclusão do *Relatório* e visa condensar em um único parágrafo as informações fáticas e legais que melhor definem o conteúdo do documento.
- **Relatório:** contém um resumo do processo original, os argumentos que fundamentaram o recurso, a 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. Ao final do relatório é apresentado o voto do relator, ou seja, a decisão que o relator propõe ao colegiado (órgão julgador do recurso).
- **Votos ou Decisão:** é a seção na qual consta a apreciação e o voto dos demais integrantes do colegiado sobre as informações apresentadas 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, a data e o local.
- **Assinatura:** nesta seção constam o nome do relator e a sua assinatura.

[34] Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/l5869.htm>. Acesso em: 10 jul. 2013.

Apresentada a estrutura fundamental do Acórdão, dá-se por finalizado esta subseção. Na próxima subseção serão vistos alguns detalhes operacionais sobre a atividade de coleta do corpus composto de Acórdãos do TJRS.

5.0.3 A coleta do corpus

Uma das primeiras definições para a implementação do estudo de caso está relacionada a escolha do corpus jurídico a ser utilizado. A primeira decisão em relação ao corpus foi que seriam utilizados documentos gerados no transcorrer do processo jurídico e não a legislação em si. Esta definição está diretamente relacionada ao objetivo de contribuir com a implementação do sistema de RI jurisprudencial definido no âmbito do Projeto CNJ Acadêmico.

O processo legal, desde a respectiva petição inicial, pode ter muitos desdobramentos, os quais geram diversos documentos, sendo então necessário definir-se qual dos documentos seria utilizado para o desenvolvimento do estudo de caso.

Por haver tratativas de parcerias entre o grupo de pesquisa SEMANTEC e um dos Tribunais Federais do sistema judiciário, decidiu-se que o documento a ser utilizado seria aquele em que se registram as decisões colegiadas dos Tribunais, ou seja, o Acórdão.

Principalmente por uma questão de proximidade geográfica, mas também pela disponibilidade para o esclarecimento de dúvidas, definiu-se que o estudo de caso seria implementado com base nos documentos disponibilizados pelo Tribunal de Justiça do Rio Grande do Sul (TJRS) no seu site de busca jurisprudencial[35].

Na Figura 19 apresenta-se a interface de pesquisa disponibilizada pelo TJRS em seu site. Foi através desta interface que se realizou a busca e coleta dos Acórdãos utilizados no estudo de caso.

Definidos o repositório e o tipo de documento, passou-se à fase da coleta do material. Objetivando-se haver uma confluência com os trabalhos desenvolvidos por integrantes do Projeto CNJ Acadêmico, definiu-se como parâmetro de busca para o *inteiro teor* dos Acórdãos os mesmos termos utilizados por Minghelli (2011), quais sejam: “apelação” e “crime”.

Figura 19: Site de busca Jurisprudencial do TJRS

[35] Disponível em: <http://www.tjrs.jus.br/busca/>. Acesso em: 10 jul. 2013.

O motivo de utilizar os mesmos parâmetros de busca usados em Minghelli (2011) relaciona-se à sua influência na modelagem da ontologia de domínio utilizada no estudo de caso. Como grande parte das classes da ontologia de domínio desenvolvida no âmbito deste trabalho derivam-se das categorias propostas por Minghelli, evidencia-se naturalmente a relevância de haver uma coerência entre os termos utilizados para a busca dos documentos utilizados em ambos os trabalhos.

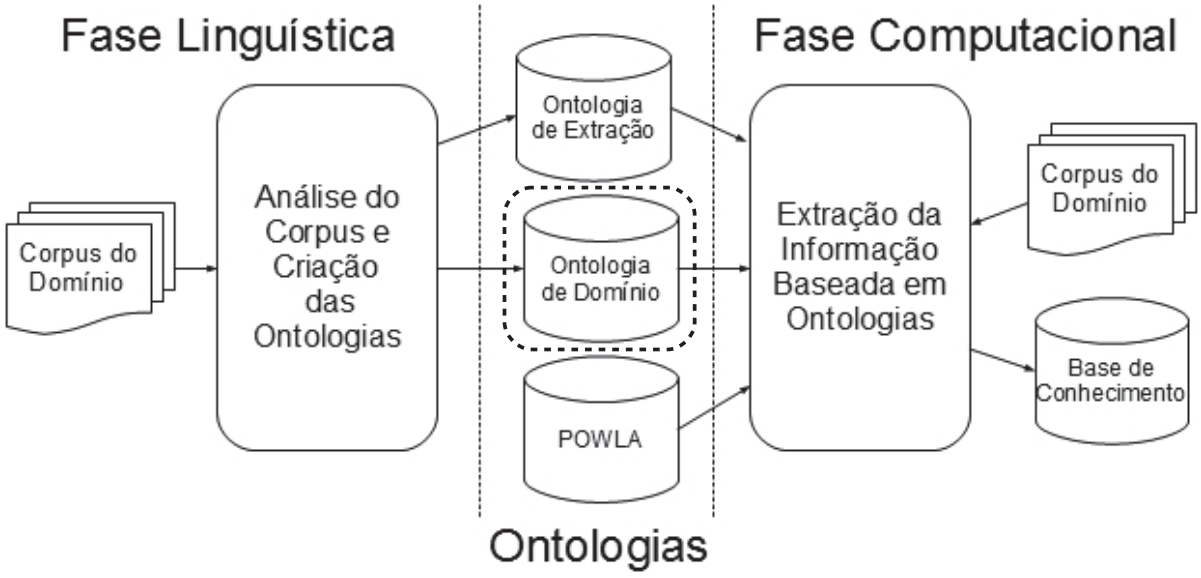
A partir do retorno da consulta, foram selecionados os primeiros 210 Acórdãos, contendo 39.895 sintagmas, totalizando 618.892 palavras, abrangendo documentos elaborados por 19 Desembargadores do TJRS. O conteúdo destes Acórdãos foram salvos individualmente em arquivos do tipo *texto puro*, uma vez que este é o formato aceito como entrada pelo parser escolhido para o desenvolvimento do estudo de caso.

Finalizada a coleta dos documentos, deu-se início a fase de criação das ontologias de domínio e de extração, ambas utilizadas no processo de EI aqui sugerido. A próxima seção relata os detalhes desta etapa.

5.1 Criação da Ontologia do Domínio Jurídico Brasileiro

A aplicação do processo de EI sugerido neste trabalho demanda a disponibilidade de uma ontologia de domínio (destacado na Figura 20 pelo retângulo tracejado). Sendo um recurso essencial para o desenvolvimento do estudo de caso, realizou-se uma busca por pesquisas anteriores referentes a ontologias na área jurídica brasileira.

Figura 20: Visão geral do modelo com destaque para a ontologia de domínio.



Fonte: elaborado pelo autor.

A partir do estudo foi implementado um protótipo de ontologia de domínio para o sistema jurídico brasileiro, a ser utilizada no estudo de caso. A estrutura desta ontologia foi fortemente influenciada pelos trabalhos desenvolvidos por Alves (2005) e Minghelli (2011).

Com base nos trabalhos de Alves e Minghelli criou-se uma ontologia de domínio jurídico básica, referenciada neste trabalho pelo acrônimo ODomJurBR (Ontologia do Domínio Jurídico Brasileiro), cuja a estrutura geral é apresentada na Figura 21.

A ontologia ODomJurBR foi formalizada na linguagem OWL através do editor de ontologias Protégé versão 4.3 (HORRIDGE et al., 2004), contendo:

- uma classe base;
- dezessete subclasses;
- 47 axiomas (dos quais 25 são lógicos); e
- cinco propriedades objeto.

Observa-se que a ontologia desenvolvida para o estudo de caso contém somente os elementos necessários e suficientes para a aplicação do processo de extração, pois não se tem como objetivo a modelagem do domínio jurídico brasileiro, mas somente a representação de alguns dos conceitos do domínio para a realização do processo de EI sugerido neste trabalho.

Figura 21: Estrutura geral da ontologia ODomJurBR.

Fonte: elaborado pelo autor.

Conquanto já se tenha justificado o motivo pelo qual a ODomJurBR contenha esta estrutura simplificada, é importante salientar-se que este trabalho também visa elaborar uma metodologia que, diferentemente das implementações conhecidas de EI baseada em ontologias, independa da terminologia utilizada na ontologia de domínio, conforme visto no capítulo anterior. A independência da terminologia permite que a estrutura da ontologia de domínio tenha representado somente os conceitos do domínio, simplificando assim a estrutura da ontologia as representações puramente lexicais.

Na próxima seção detalha-se os procedimentos adotados para a criação da outra ontologia de extração a ser utilizada neste primeiro estudo de caso, encerrando-se assim a Fase Linguística do modelo.

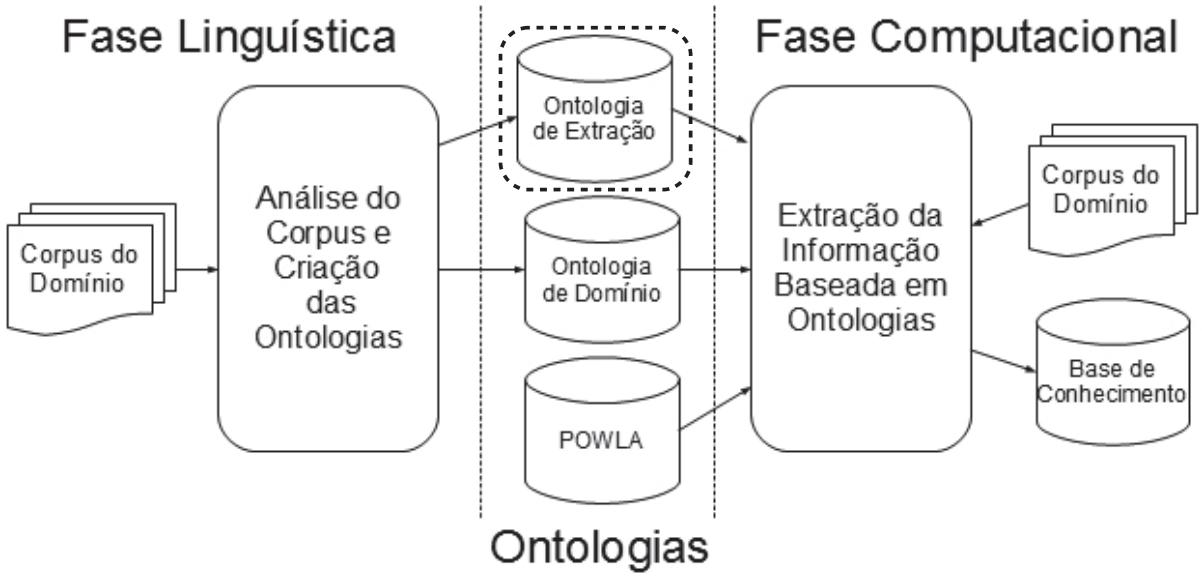
5.2 Criação da Ontologia de Extração

Conforme modelo proposto neste trabalho, a ontologia de extração, destacada no retângulo tracejado da Figura 22, é elaborada pelo engenheiro de conhecimento a partir da identificação de padrões que compõem e/ou circundam os conceitos e relacionamentos ontológicos nos sintagmas previamente identificados pelos linguistas.

No estudo de caso aqui desenvolvido, optou-se pelo uso de informações linguísticas como elemento caracterizador dos sintagmas do domínio para a composição das regras. Desta forma, sendo o corpus formado por documentos do sistema jurídico brasileiro, torna-se necessário o uso de um *parser* linguístico que analise textos escritos em português.

Devido ao bom desempenho apresentado na análise dos documentos jurídicos, reforçado pela posse de licença de uso por parte do grupo de pesquisa SEMANTEC, optou-se pelo parser linguístico *Palavras* (BICK, 2000) para o desenvolvimento do estudo de caso.

Figura 22: Visão geral do modelo com destaque para a ontologia de extração.



Fonte: elaborado pelo autor.

Os procedimentos adotados na etapa de identificação dos padrões linguísticos deste estudo de caso seguiram a metodologia exposta na seção 4.1.2, tendo sido aleatoriamente escolhidos 10 documentos do corpus para a análise de corpus na Fase Linguística.

A realização da análise de corpus resultou na identificação de diversos sintagmas que faziam referência aos eventos jurídicos da ODomJurBR. Estes sintagmas foram submetidos ao parser Palavras e os respectivos padrões linguísticos utilizados para a formalização das regras linguísticas na ontologia de extração.

5.2.1 Formalização das Regras Linguísticas de Extração

A EI implementada através de regras de extração em geral tem seu funcionamento baseado em padrões que são avaliados para a identificação de referências a elementos de interesse nos documentos. No caso de EI baseadas em ontologias, os elementos de interesse são geralmente referências textuais aos conceitos e relacionamentos descritos na ontologia.

Conforme visto anteriormente, uma vez que as informações linguísticas dos documentos estejam representadas no formato OWL, torna-se possível utilizar o próprio sistema inferencial da ontologia para a representação das regras de extração.

No contexto do estudo de caso implementado neste trabalho, as RACs são regras de extração, formalizadas em axiomas da Lógica de Descrição (DL) ou regras da linguagem Semantic Web Rule Language (SWRL), que verificam as características linguísticas de frases para identificar condições semânticas que permitam associar os conceitos da ontologia ODomJurBR aos termos ou trechos do documento jurídico.

A ontologia de extração, a qual contém as RACs deste estudo de caso, foi denominada de ODomJurBR-RAC. Esta ontologia, somada à ontologia de domínio jurídico ODomJurBR e ao modelo de dados POWLA, compõe o grupo de ontologias necessárias para a implementação do processo de EI dos Acórdãos do corpus.

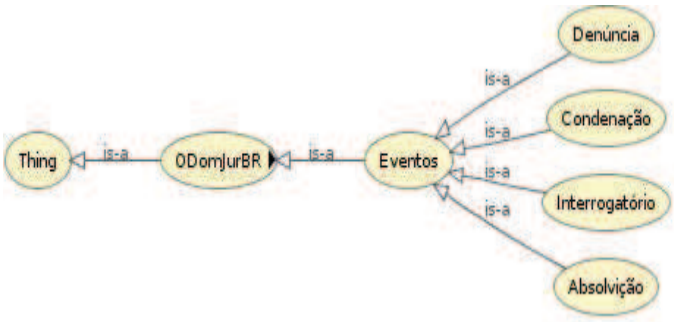
Para o desenvolvimento deste estudo de caso, alguns eventos do domínio jurídico foram selecionados para serem representados na ontologia ODomJurBR, como pode ser visto na Figura 23. A escolha dos eventos a serem identificados no corpus está diretamente ligada à frequência com que estes eventos aparecem nos Acórdãos do TJRS.

Para identificar-se a presença dos eventos legais nos documentos é necessário definir-se um ponto de partida para a sua localização. Uma abordagem possível para a identificação da presença de eventos em um sintagma seria a busca do termo que comumente o representa, tanto na forma verbal quanto na nominal.

Quadro 3: Exemplos de frases que referenciam eventos jurídicos.

1) "Na Comarca de Xxxxxxx, o Ministério Público **denunciou** XXXX XXXXXXXXX XXX XXXXXX como incurso nas sanções do artigo 121, § 2º, inciso IV, do Código Penal, ..."
2) "Na Comarca de Xxxxxxxxx Xxxxxxxxxx, o Ministério Público ofereceu **denúncia** contra Xxxx Xxxxxx xx Xxxxx, dando-o como incurso nas sanções do artigo 14, caput, da Lei nº 10.826/03, ..."

Figura 23: Classe *Eventos* da ontologia ODomJurBR



Fonte: elaborado pelo autor.

No trecho de texto retirado de um dos Acórdãos do corpus e apresentado no item 1 do Quadro 3, pode-se observar uma referência ao evento *Denúncia* expressada pelo uso do verbo *denunciar*. A outra possibilidade de referência ao evento seria pela sua forma nominal "*denúncia*", como ilustrado no item 2 do mesmo quadro.

Um sistema de EI que busque a localização de referência a *Eventos* jurídicos deve identificar nestas duas sentenças a presença do evento *Denúncia*, pois em ambas as sentenças nota-se a presença do Ministério Público como agente da ação.

Nos próximos parágrafos será apresentado o processo de criação e formalização de uma regra linguística de extração para a identificação do evento jurídico *Denúncia,* utilizando-se a acepção conceitual dada no dicionário jurídico de Silva (2010): *Denúncia* é um termo que possui aplicação nos Direitos Civil, Penal ou Tributário, sempre com o significado genérico de declaração feita em juízo ou notícia de fato que deva ser comunicado.

No estudo de caso interessa o significado específico do Direito Penal, onde a denúncia é, em sentido estrito, "o ato mediante o qual o representante do Ministério Público formula sua acusação perante o juiz competente a fim de que se inicie a ação penal contra a pessoa a quem se imputa a autoria de um crime ou de uma contravenção" (SILVA, 2010).

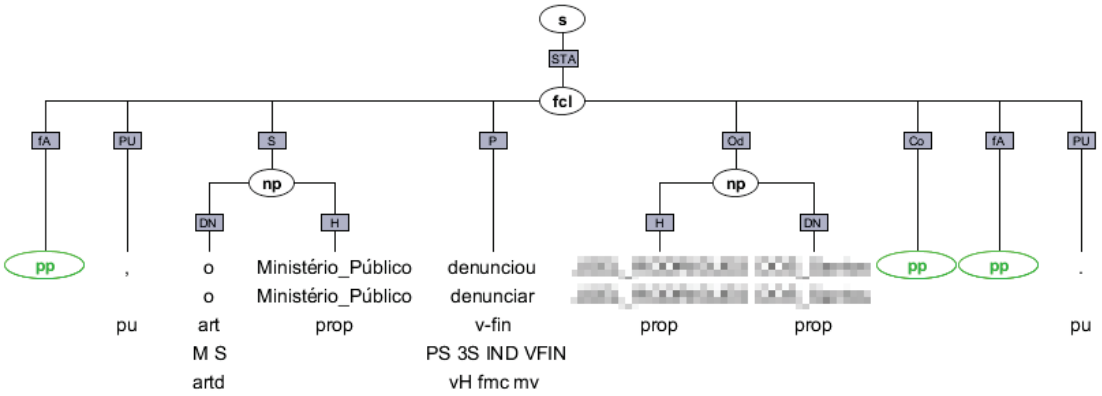
Na definição do evento dada acima, observa-se alguns componentes periféricos da *Denúncia*: o Ministério Público (entidade jurídica), o promotor (representante do Ministério Público), o juiz, o réu (ou denunciado), o ato ilícito (crime ou contravenção).

Observa-se também a presença de alguns relacionamentos: o agente da denúncia no contexto do Direito Criminal é obrigatoriamente um representante do Ministério Público; o ato ilegal deve sempre ser submetido à apreciação de um juiz que seja competente para julgá-lo; no polo passivo da denúncia está o réu; o fundamento da denúncia é um ato ilegal (ou fato típico), devidamente descrito em uma lei.

Tem-se então uma série de elementos que orbitam o evento jurídico *Denúncia*. Uma frase de um Acórdão, no contexto do Direito Penal, que contenha o verbo d*enunciar* é forte candidata a conter referência ao evento *Denúncia*. No entanto, somente a presença do verbo *denunciar* não é suficiente para concluir-se que há uma referência ao evento de *Denúncia* criminal, conforme as definições associadas ao conceito.

Torna-se imprescindível avaliar outros elementos da frase para se obter uma maior precisão quanto a presença ou não da referência ao conceito *Denúncia*. Será com base nas

Figura 24: Árvore de sintaxe com o verbo Denunciar.



Fonte: elaborado pelo autor.

informações linguísticas da frase que serão obtidos outros indicativos que asseguram a certeza da referência ao conceito ontológico.

A Figura 24 apresenta algumas das informações linguísticas geradas pelo analisador Palavras para a frase apresentada no item 1 do Quadro 3. São mostradas somente as informações mais relevantes para a elaboração da RAC, sendo ocultadas as demais informações.

O verbo do sintagma apresentado na Figura 24 está identificado sob a aresta da árvore etiquetada com a letra "P" (predicativo). Pode-se também observar as informações morfológicas, gramaticais e semânticas que o Palavras agregou ao termo.

A etiqueta semântica "*vH fmc mv*" indica que o termo "*denunciou*" é um verbo que demanda um ser humano como agente (*verb human* - "vH"), sendo o verbo principal (*main verb* - "*mv*") do sintagma verbal finito (*finite main clause* - "*fmc*").

Muito embora tais informações não sejam necessárias para a identificação do evento *Denúncia*, é importante frisar que tais subsídios semânticos estão à disposição para a elaboração das RACs, permitindo verificar-se inclusive a coerência da frase.

Contudo, é a partir das informações sintáticas geradas pelo parser que será identificado o agente do verbo, pela localização do sujeito (elemento etiquetado com "*S*") do sintagma finito (*finite clause* - "*fcl*") na árvore de sintaxe.

Pode-se ainda verificar-se na frase da Figura 24 que o sujeito da frase analisada é um sintagma nominal (*noun phrase* - "*np*") composto por dois elementos: o adjunto adnominal ("*DN")* e o núcleo do sujeito (*head* - "*H")*.

Então, para que a frase contenha uma referência ao evento *Denúncia,* é imprescindível verificar se o núcleo do agente do verbo *denunciar* é um representante do *Ministério Público*, pois somente um representante desta entidade jurídica pode aparecer como polo ativo de uma denúncia criminal.

Na frase apresentada como exemplo na Figura 24, observa-se que o parser Palavras reconheceu os termos "*Ministério*" e "*Público*" como uma informação única que representa o

nome de uma entidade, sendo por isto ambos os termos classificados gramaticalmente como um único substantivo próprio ("*prop*").

Analisando-se a frase ("*fcl*") que contém o verbo *denunciar* ("*P*"), verifica-se que o núcleo ("*H*") do sintagma sujeito ("*S*") é realmente o *Ministério Público*, configurando-se assim uma referência inequívoca a ocorrência do evento *Denúncia*.

Os elementos linguísticos citados no parágrafo acima são as características que, presentes em um sintagma, denotariam a presença do evento. Resta agora formalizar a RAC que utiliza estas informações linguísticas para a efetiva identificação do evento jurídico.

Relembrando o processo da metodologia computacional, o documento a ser submetido ao processo de EI é um arquivo no formato texto puro submetido ao *parser* para a inserção de informações abstratas, as quais são posteriormente representadas no formato POWLA.

A saída produzida pelo *parser* linguístico Palavras não é um arquivo OWL, mas sim um arquivo-texto contendo o documento linguisticamente anotado em um formato chamado *árvore deitada* (BICK, 2000). O Quadro 6 apresenta como ficam representadas no formato *árvore deitada* as informações linguísticas vistas na Figura 24.

Não se encontra ainda disponível um sistema que converta diretamente do formato *árvore deitada* para o formato OWL, sendo então necessário primeiro converter-se a saída do *parser* para um formato intermediário.

O formato TIGER-XML foi a solução encontrada para esta questão. Dentre as diversas ferramentas que acompanham o *parser* Palavras há um script, desenvolvido em *Perl* [36], que converte a saída do parser para TIGER-XML. O Quadro 5 apresenta como ficam representadas em TIGER-XML as informações linguísticas vistas no Quadro 6.

Verifica-se que a maior parte das informações linguísticas geradas pelo Palavras no formato *árvore deitada* estão presentes na representação em TIGER-XML. Mas é importante notar-se também que algumas das informações linguísticas são perdidas no processo de conversão.

Alguns exemplos de informação linguística presentes na *árvore deitada* mas omitidas na versão TIGER-XML: o artigo definido "*o*" é o sexto elemento da frase ("*#6*"), sendo um determinante para o próximo substantivo à sua direita ("*@>N*"); este substantivo é o sétimo elemento da frase ("*#6->7*").

Quadro 4: Informações linguísticas no formato *Árvore Deitada*

```
...
o                    [o] <artd> DET M S @>N  #6->7
Ministério=Público   [Ministério=Público] < PROP M S @SUBJ>  #7->8
denunciou            [denunciar] <vH> <fmc> <mv> V PS 3S IND VFIN @FS-STA  #8->0
...
```

[36] <http://www.perl.org/>. Acesso em: 30 out. 2013.

Quadro 5: Informações linguísticas no formato TIGER-XML.

```
...
<terminals>
...
<t id="s1_5"
  word="o" lemma="o" pos="art" morph="M S" extra="artd"
/>
<t id="s1_6"
  word="Ministério_Público" lemma="Ministério_Público" pos="prop" morph="--" extra="cjt-S"
/>
<t id="s1_7"
  word="denunciou" lemma="denunciar" pos="v-fin" morph="PS 3S IND VFIN" extra="vH fmc mv"
/>
…
</terminals>
<nonterminals>
…
 <nt id="s1_501" cat="np">
            <edge label="DN" idref="s1_5"/>
            <edge label="H" idref="s1_6"/>
</nt>
…
</nonterminals>
```

Outra omissão percebida: o "*Ministério=Público*" é o sétimo elemento da frase ("*#7*") e cumpre com o papel sintático de sujeito do verbo que está à direita ("*@SUBJ>*"), sendo este verbo o oitavo elemento da frase ("*#7->8*").

Em situações específicas observadas no decorrer do estudo de caso, como por exemplo ao aplicar-se as regras linguísticas à frases com estrutura sintática muito complexa, observou-se que a perda destas informações tem como efeito uma imprecisão semântica para a regra linguística. O impacto desta perda de informações linguísticas sobre as RACs será visto novamente no capítulo 6.

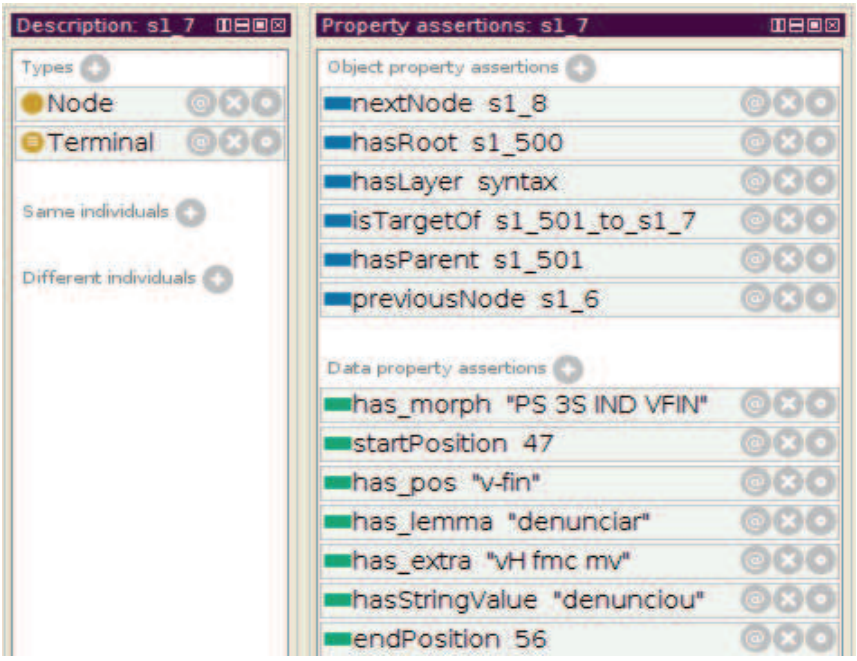
Uma vez representadas as informações linguísticas no formato TIGER-XML, torna-se possível convertê-las para OWL através da submissão do arquivo a um *script* XSLT (CLARK, 1999) desenvolvido a partir do conversor *tiger2owl.xsl*[37].

Devido a problemas de incompatibilidade entre o TIGER-XML gerado pelo Palavras e o formato esperado pelo *script tiger2owl.xsl*, foi necessário desenvolver um script conversor que tratasse das especificidades do arquivo gerado pelo Palavras.

Após um trabalho de depuração do script original, os problemas foram localizados e sanados, dando origem a um novo conversor chamado *palavras2owl.xsl*, o qual então converte as anotações linguísticas em TIGER-XML geradas pelo Palavras para o formato OWL, utilizando o POWLA como modelo de dados.

[37] <http://sourceforge.net/p/powla/code/14/tree/trunk/tools/xslt/tiger2owl.xsl>. Acesso em: 11 Jul. 2013.

Figura 25: Resultado da conversão das informações linguísticas para OWL



Fonte: elaborado pelo autor.

A Figura 25 apresenta um exemplo de como ficam representadas algumas das informações linguísticas do Palavras na linguagem OWL. O nó s1_7 da árvore de sintaxe do TIGER-XML originou o indivíduo s1_7 da classe *Terminal* na ontologia POWLA.

Verifica-se que as *Propriedades Objeto* do *indivíduo* permitem identificar a sua localização em relação à árvore sintática da frase, descrevendo as relações existentes com os demais nodos da árvore, enquanto que as *Propriedades de Dados* do *indivíduo* descrevem as informações linguísticas do termo geradas pelo *parser*.

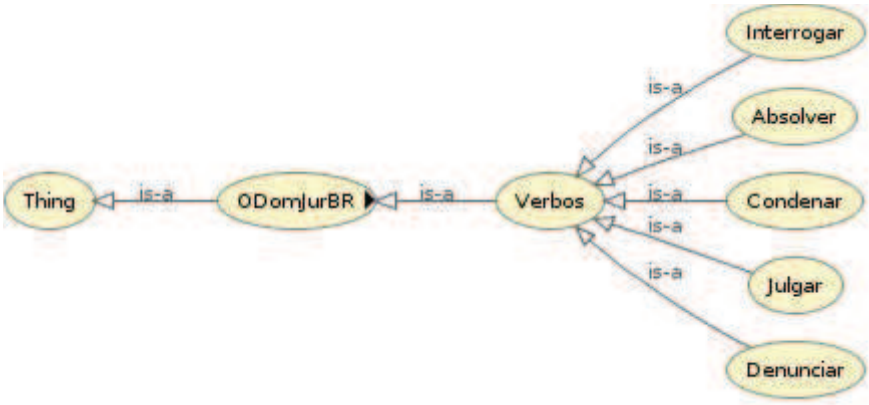
São estas informações linguístico estruturais, disponíveis agora no formato OWL, que serão utilizadas para a formalização das RACs da ontologia de extração ODomJurBR-RAC. Concretamente, a ODomJurBR-RAC é uma ontologia expressa em OWL, composta pela importação dos conceitos da ontologia de domínio ODomJurBR e das RACs formalizadas em DL ou SWRL. A principal função da ODomJurBR-RAC é estabelecer uma associação direta entre trechos do documento representado em OWL e os conceitos do domínio, associação esta expressa por regras linguísticas representadas nas RACs.

A fim de melhor esclarecer como ocorre esta associação entre conceitos ontológicos e trechos dos documentos, serão formalizadas na próxima subseção as RACs necessárias para a identificação do evento *Denúncia*.

5.2.2 Formalização de RACs em DL

Visando principalmente a didática, será vista agora a formalização de uma das regras de extração mais simples da ODomJurBR-RAC: a busca por referências ao conceito expresso

Figura 26: Classe Verbos da Ontologia ODomJurBR.



Fonte: elaborado pelo autor.

pela classe *Denunciar* da ODomJurBR. Como pode ser visto na Figura 26, a classe *Denunciar* é uma subclasse do conceito *Verbo*, ou seja, representa o verbo *denunciar*[38].

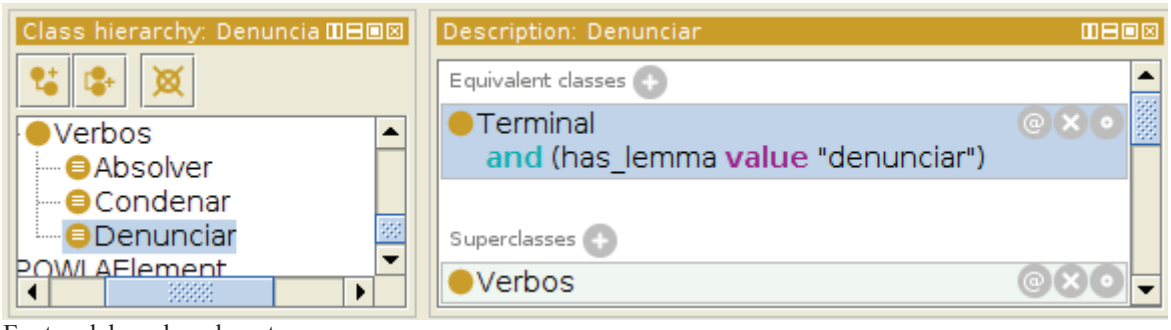
A regra linguística para identificação das ocorrências da classe *Denunciar* é bastante simples: todos os nós terminais do documento analisado cuja a forma canônica (*lemma*) seja "*denunciar*" são indubitavelmente exemplos desta classe.

Na ontologia ODomJurBR-RAC, a qual herdou via importação os conceitos da ODomJurBR, será então acrescentado um axioma DL de equivalência para a classe *Denunciar*, o qual tem como objetivo representar a regra que define as características linguísticas dos indivíduos que a representam, como pode ser visto na Figura 28.

Verifica-se que o indivíduo *s1_7*, apresentado na Figura 25 encaixa-se na descrição: ele é um indivíduo da classe *Terminal* e tem uma *Propriedade de Dados* chamada *has_lemma*, cujo conteúdo é a sequência de caracteres "*denunciar*".

Reunidas as ontologias ODomJurBR, ODomJurBR-RAC e a representação em POWLA da frase apresentada no item do Quadro 3 em um único sistema inferencial, como

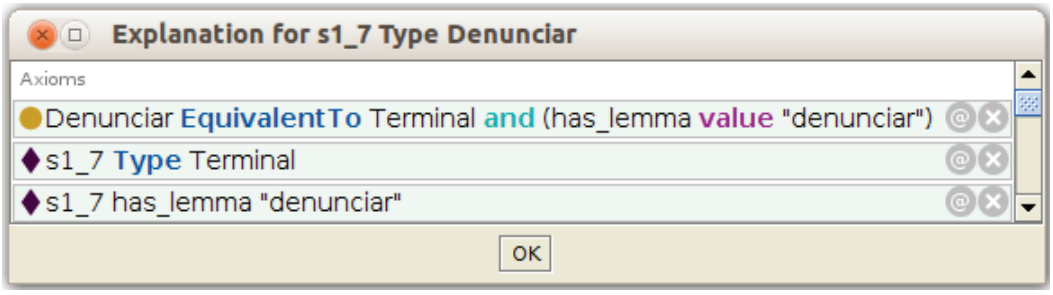
Figura 28: RAC linguística para identificação do verbo denunciar.



Fonte: elaborado pelo autor.

[38] A classe *Verbos* foi inserida na ontologia ODomJurBR principalmente por influência do trabalho de Alves (2005), não sendo objetivo deste trabalho questionar ou justificar a inserção deste conceito na ontologia de domínio jurídico brasileira.

Figura 27: Justificativa para a inferência de que s1_7 é um indivíduo da classe Denunciar



Fonte: elaborado pelo autor.

por exemplo o editor de ontologias Protégé, é possível verificar-se por lógica inferencial que o indivíduo *s1_7* é uma *instância* (uma ocorrência, um exemplo) da classe *Denunciar.*

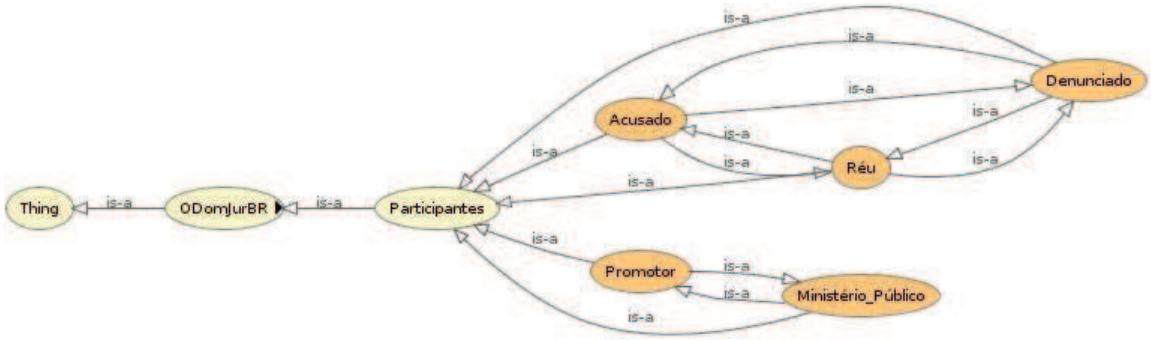
A Figura 27 apresenta, a título de exemplo, a justificativa dada pelo *reasoner* Pellet para explicar o porquê da sua conclusão de que o indivíduo *s1_7* pertence ou representa a classe *Denunciar*. Está assim elaborada a primeira RAC linguística de extração, formulada através de um axioma DL, cujo objetivo é identificar elementos textuais de documentos que representam o verbo *denunciar*.

Contudo, para a identificação do evento *Denúncia* faz-se ainda necessário a realização da verificação de outros elementos na frase, pois a presença do verbo *denunciar* é somente um dos indicativos da referência a este evento jurídico.

Uma das formas de se garantir que o verbo *denunciar* está referenciando a ocorrência do evento *Denúncia* no contexto criminal é através da confirmação de que o agente do verbo é um representante do Ministério Público, pois, no sistema jurídico brasileiro, a denúncia crime somente pode ser representada perante a justiça pelo próprio Estado, através de um representante do Ministério Público.

Torna-se então necessário verificar-se a presença do *Ministério Público* como agente do verbo *denunciar*. Pode-se observar na Figura 29 que a entidade jurídica Ministério Público é representada como um subtipo de *Participantes.* A classe *Participantes* reúne sob si os papéis que cada indivíduo do mundo real exerce no domínio jurídico brasileiro.

Figura 29: Subclasse Ministério_Público da ODomJurBR.



Fonte: elaborado pelo autor.

Observe-se que, além da definição de *Ministério_Público* como um tipo de *Participantes*, há também uma relação de equivalência entre as classes *Ministério_Público* e *Promotor*, assim como entre as classes *Réu*, *Acusado* e *Denunciado*, denotando uma relação semântica de sinonímia entre estes papéis do sistema jurídico brasileiro na ODomJurBR.

A abordagem de EI sugerida neste trabalho leva em conta estas relações, fazendo uso destes conhecimentos específicos da ontologia de domínio no momento de avaliar as regras linguísticas de extração.

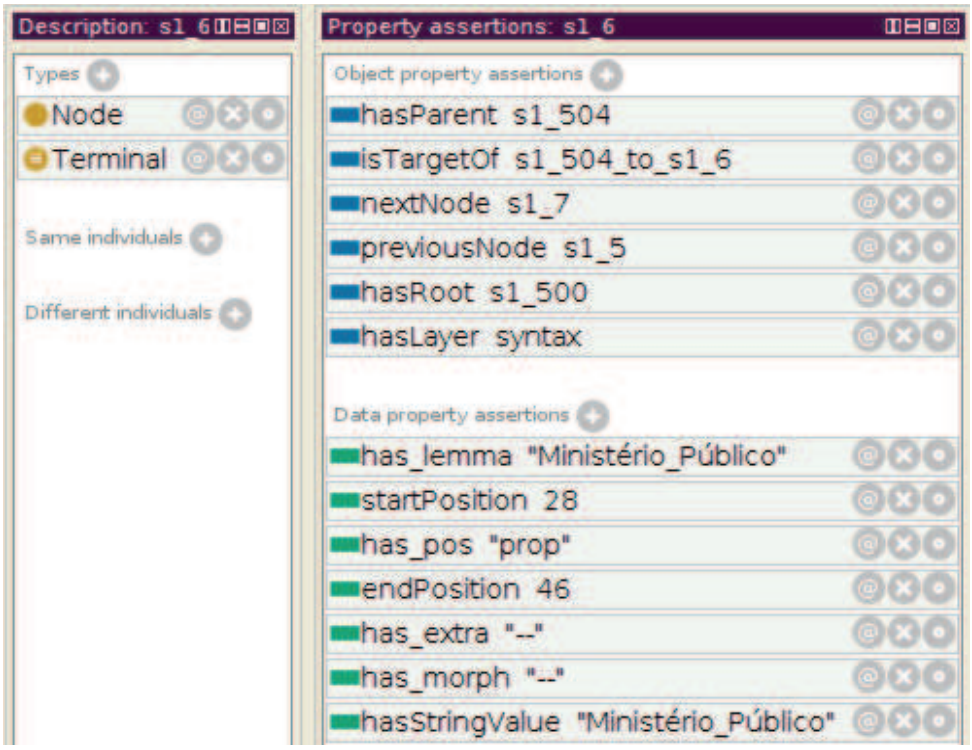
Esta reutilização dos conceitos ontológicos do domínio é viabilizada pelo uso da lógica unificada proposta neste trabalho. Regras linguísticas que exploram as relações de sinonímia serão vistas novamente mais adiante, ainda neste capítulo.

Voltando à regra linguística para identificação da entidade Ministério Público, esta é igualmente simples, muito facilitada pelo reconhecimento automático de entidades que o *parser* Palavras realiza na sua análise.

Como pode ser visto na Figura 31, os termos "*Ministério"* e "*Público"* são classificados como um único termo (*has_lemma* = "*Ministério_Público*"), gramaticalmente classificados como substantivo próprio (*has_pos* = "*prop*"), pois representam uma entidade nomeada do mundo real.

Fundamentalmente, a regra linguística para identificação do *Ministério Público* resume-se a verificação da forma canônica do termo através de uma comparação simples do conteúdo da propriedade de dados *has_lemma*.

Figura 31: Referência ao Ministério Público em OWL.



Fonte: elaborado pelo autor.

Figura 30: Axioma DL para identificação da entidade Ministério Público.

Fonte: elaborada pelo autor.

O reconhecimento da entidade *Ministério Público,* realizado pelo parser, será aproveitado na definição da RAC linguística para a identificação do *Ministério Público*, resultando em um axioma DL tão simples quanto o utilizado para a identificação do verbo *denunciar*, como pode ser visto na Figura 30.

Criadas as regras que caracterizam o verbo *denunciar* e o participante *Ministério Público*, pode-se agora partir para a elaboração da regra que identifica a referência ao evento *Denúncia*.

5.2.3 Formalização de RACs em SWRL

Relembrando novamente as características linguísticas dos sintagmas que se referem à denúncia: a frase deve conter o verbo *denunciar* como *predicativo*, sendo o sujeito (ou seja, o agente da ação expressa pelo verbo) a entidade *Ministério Público*.

A RAC para identificar se um sintagma contém uma referência ao evento *denúncia* é mais complexa do que as anteriormente vistas. O evento exige a verificação da função sintática dos elementos que compõem o sintagma. Por este motivo a formalização desta regra será feita utilizando-se a linguagem SWRL.

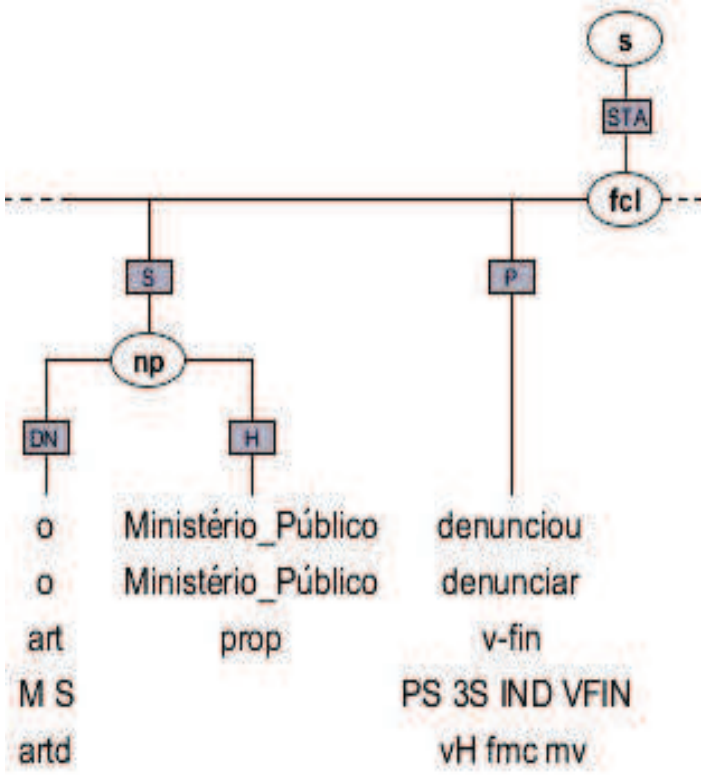
Um exemplo de RAC em SWRL para identificação de um elemento da classe *Denúncia* é apresentada no Quadro 6. Já na linha 1 verifica-se a reutilização da regra em DL elaborada para a identificação do verbo *Denunciar*. Aqui a função desta regra é verificar a presença do verbo *denunciar* na frase sendo analisada. Caso seja encontrada uma ocorrência do verbo, associa-se esta referência à variável "*?verbo"*.

Encontrado o verbo *denunciar*, busca-se na árvore sintática o sintagma que o contém (linha 2), pois é necessário verificar se o sujeito (agente da ação) é um indivíduo da classe *Ministério_Público*. Considerando-se as informações apresentadas na Figura 32, vê-se que o nó etiquetado com *fcl* é também o nó *pai* do verbo *denunciar*. A *Propriedade Objeto hasParent* do *indivíduo* representado pela variável *?verbo* contém a referência ao nó *pai* do verbo *denunciar*, a qual é armazenada na variável *?fcl*.

Quadro 6: RAC para identificação do evento Denúncia, formalizada em SWRL.

```
1. Denunciar(?verbo),
2. hasParent(?verbo, ?fcl),
3. isSourceOf(?fcl, ?rel),
4. has_label(?rel "S"^^string),
5. hasTarget(?rel, ?suj),
6. hasChild(?suj,?mp)
7. Ministério_Público(?mp)
8.  -> Denúncia(?fcl)
```

Figura 32: Árvore de sintaxe para a elaboração da RAC *Denúncia*.



Fonte: elaborado pelo autor.

O sintagma representado pelo nó *?fcl* tem diversos nodos *filhos*, sendo necessário encontrar-se o nó filho que cumpra com o papel de sujeito da frase. A localização do sujeito se dá pela busca de uma relação cuja origem seja o sintagma "*?fcl*" (linha 3) e o destino seja um elemento marcado com a etiqueta "S" (linha 4).

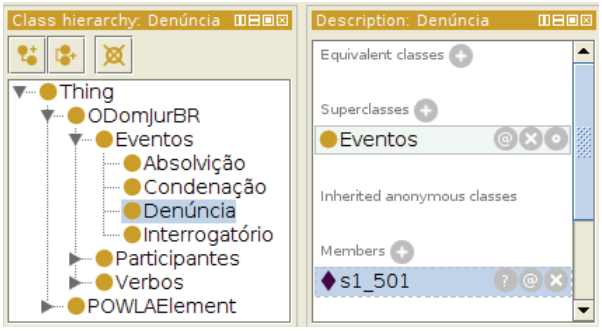
Encontrada a relação que satisfaça esta condição, guarda-se na variável "*?suj*" (linha 5) a referência a este elemento. Observando-se novamente a Figura 32, verifica-se que o sujeito da frase que contém o verbo *denunciar* é um sintagma nominal que contém dois elementos terminais: "*o*" e "*Ministério_Público*".

O próximo passo é verificar se um dos elementos que compõem o sujeito da frase é um representante da classe *Ministério_Público*. Isto é realizado pela análise dos filhos do sintagma "*?suj*" na árvore sintática (linha 6).

Se no *sujeito* da frase houver a ocorrência do conceito *Ministério_Público* (linha 7), então é possível definir-se que o sintagma que contém o sujeito e o predicado da frase (referenciado na variável "*?fcl*") é uma instância da classe *Denúncia* (linha 8).

Estão assim formuladas as três RACs necessárias para a identificação do conceito expresso pela classe *Denúncia*: duas regras formalizadas em linguagem DL, para a identificação de referências ao verbo *Denunciar* e ao participante *Ministério_Público*, e uma terceira regra formalizada em SWRL para a verificação de ocorrência do evento *Denúncia*.

Figura 33: Evento Denúncia localizado no nó s1_501.



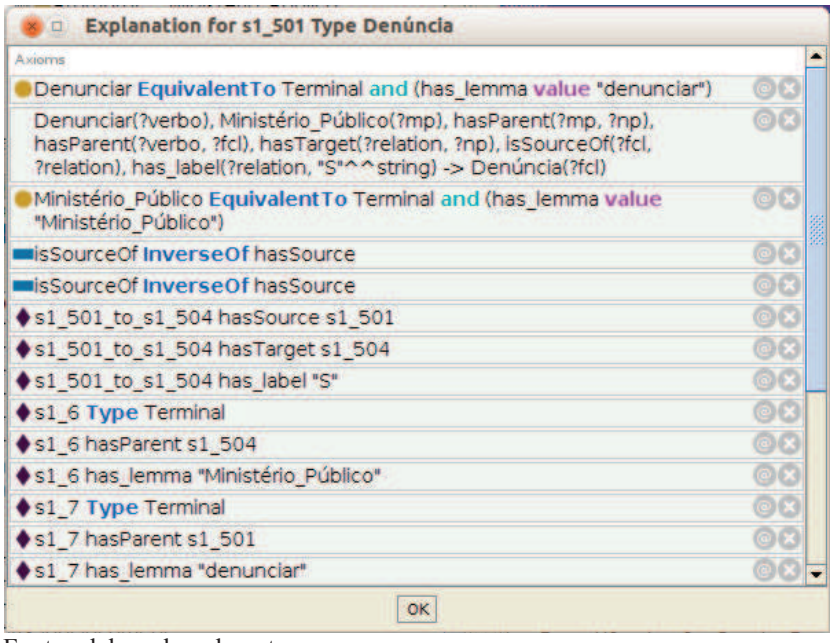
Fonte: elaborado pelo autor.

Formalizadas estas RACs na ODomJurBR-RAC, basta que sejam carregadas e mescladas todas as ontologias do modelo para que, aplicado o *reasoner*, seja concluído por inferência lógica que o sintagma (indivíduo) *s1_501* é uma instância da classe *Denúncia,* ou seja, o sintagma apontado pelo nó *s1_501* contém uma ocorrência do evento jurídico *Denúncia*, como demonstrado na Figura 33.

Solicitando-se novamente que seja justificado o motivo pelo qual o *reasoner* inferiu a relação entre o indivíduo *s1_501* e a classe *Denúncia,* obtém-se como retorno uma extensa lista que demonstra todos os elementos lógicos considerados no raciocínio, conforme ilustrado na Figura 34.

Está assim demonstrado o processo adotado neste estudo de caso para a formalização das regras de extração baseada em informações linguísticas que utilizam a capacidade inferencial da linguagem OWL para a identificação de conceitos ontológicos em documentos.

Figura 34: Explicação do reasoner para a inferência do evento Denúncia.



Fonte: elaborado pelo autor.

5.2.4 Facilitando a Pesquisa por Nodos na Árvore de Sintaxe

Além das RACs de extração, é interessante destacar-se que foram também definidas na ODomJurBR-RAC duas propriedades objeto que tem como objetivo facilitar a busca de nodos na árvore de sintaxe: *hasAncestral* e *hasDescendent*.

A *propriedade hasAncestral* foi formalizada através da definição da regra SWRL apresentada no Quadro 7. Pode-se ver que a definição é bastante simples, fazendo uso da *propriedade* existente *hasParent* (linha 1) do POWLA, a qual tem a função de relacionar os nodos filhos aos respectivos pais.

Quadro 7: Definição da propriedade hasAncestral em SWRL

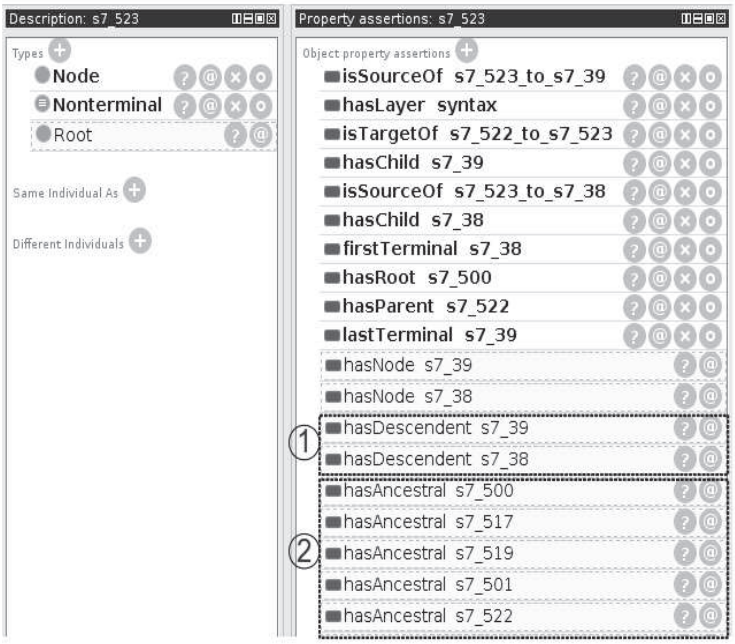
```
1. hasParent(?nodoFilho, ?nodoPai)
2. -> hasAncestral(?nodoFilho, ?nodoPai)
```

A *propriedade hasAncestral* diz que se um *nodoFilho* é filho de um *nodoPai*, então *nodoPai* é ancestral de *nodoFilho*. Definiu-se na ODomJurBR-RAC que a propriedade *hasAncestral* é *transitiva*, ou seja, é uma relação que se assume como verdeira também para os nodos descendentes do *nodoFilho*, criando-se assim uma sequência de ancestralidade.

Definida a propriedade *hasAncestral*, ao disparar o reasoner obtém-se como resultado uma lista de propriedades que permitem identificar os nodos "*ancestrais*" de um determinado nodo na árvore de sintaxe, como pode ser observado no destaque (2) da Figura 35.

A implementação da propriedade *hasDescendent*, por sua vez, é ainda mais simples, pois é necessário somente defini-la como a propriedade inversa de *hasAncestral*. Definida

Figura 35: Propriedades hasDescendent e hasAncestral



Fonte: elaborado pelo autor.

desta forma, encontra-se por inferência lógica todos os descendentes de um determinado nodo (destaque 1 da Figura 35).

Estas duas propriedades tem como função simplificar a elaboração de RACs mais genéricas, que precisam localizar elementos linguísticos em uma subárvore da árvore de sintaxe, como por exemplo a RAC para identificação do evento *Absolvição* apresentada no Quadro 8. Observa-se na linha 6 o uso da propriedade *hasDescendent*, cuja função é a busca de uma referência ao *Acusado* na subárvore que cumpre com o papel sintático de *objeto direto* (linha 4) do verbo *Absolver* (linha 1).

Para uma melhor compreensão da RAC apresentada no Quadro 8, poderia-se descrevê-la da seguinte forma: encontrado no sintagma o verbo *absolver* (linha 1) e verificada a presença de um *acusado* (linha 7) em qualquer ponto da subárvore que representa *objeto direto* do verbo (linha 4), então conclui-se haver neste sintagma a ocorrência do evento *Absolvição* (linha 8).

Quadro 8: Definição da RAC para o evento *Absolvição*.

```
1. Absolver(?tVerbo),
2. hasParent(?tVerbo, ?cl),
3. isSourceOf(?cl, ?relOd),
4. has_label(?relOd,"Od"^^string),
5. hasTarget(?relOd, ?npAcusado),
6. hasDescendent(?npAcusado,?tAcusado),
7. Acusado(?tAcusado)
8. -> Absolvição(?cl)
```

O uso da propriedade *hasDescendent* na RAC apresentada no Quadro 8 tem como efeitos: (1) a simplificação da regra, fazendo uso do algoritmo de inferência lógica do *reasoner* para a realização da busca por informações na árvore de sintaxe do sintagma; (2) a generalização da RAC, pois esta mesma regra busca o *Acusado* em qualquer parte da subárvore do *objeto direto,* permitindo localizar o elemento de interesse em sintagmas que contenham outros componentes linguísticos, tais como adjetivos, superlativos, etc.

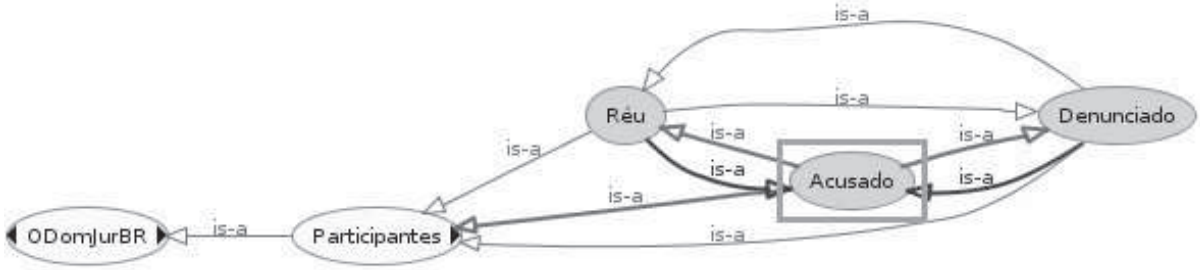
Assim implementada, esta mesma RAC identifica a presença do evento *Absolvição* nas frases "*a sentença absolveu o acusado*" e "*o juiz absolveu o injustamente acusado*", pois em ambas ocorre a presença do *Acusado* no objeto direto do verbo *Absolver*.

5.2.5 Uso das relações da Ontologia de Domínio no Processo de Extração

O exemplo apresentado no Quadro 8 será reutilizado para a demonstração de outra característica muito interessante do sistema de regras de extração baseado nas linguagens lógicas da ontologia (as RACs): o reaproveitamento de definições conceituais da ontologia de domínio para a extração de informação.

Será demonstrado o uso da sinonímia através da definição de equivalência entre as classes da ontologia de domínio. Relembrando, na ODomJurBR verifica-se a presença do uso do recurso de classe equivalente para a definição de que, no contexto jurídico, *Acusado* é sinônimo de *Réu*, que por sua vez é sinônimo de *Denunciado*, conforme pode ser visto na Figura 36.

Figura 36: Definição de sinonímia entre Acusado, Réu e Denunciado na ODomJurBR.



Fonte: elaborado pelo autor.

Uma vez que a definição de equivalência entre as classes *Acusado*, *Réu* e *Denunciado* são carregadas no momento em que a ontologia ODomJurBR é combinada com as outras três ontologias do modelo, tais definições são consideradas pelo *reasoner* no momento de avaliar as RACs.

Resulta daí que as frases "absolveu o réu" ou "absolveu o denunciado" são ambas identificadas pela RAC apresentada no Quadro 8 como contendo a ocorrência do evento *Absolvição*, pois a definição de equivalência entre *Acusado*, *Réu* e *Denunciado* permite esta generalização da regra a partir dos conceitos da ontologia de domínio.

Seguindo-se a metodologia descrita nesta seção foram formalizadas 11 RACs na ODomJurBR-RAC, baseado na análise de um corpus composto por 10 Acórdãos, abrangendo documentos elaborados por 6 Desembargadores do TJRS. Este corpus foi utilizado tanto para a identificação da ocorrência dos eventos jurídicos representados na ODomJurBR quanto para a caracterização das RACs de extração, finalizando-se assim a Fase Linguística do estudo de caso.

Conforme o modelo proposto, segue-se a Fase Computacional, etapa em que as ontologias de domínio e de extração são combinadas com os documentos representados no formato POWLA para a realização do processo de EI.

5.3 Conversão do Corpus para o formato OWL

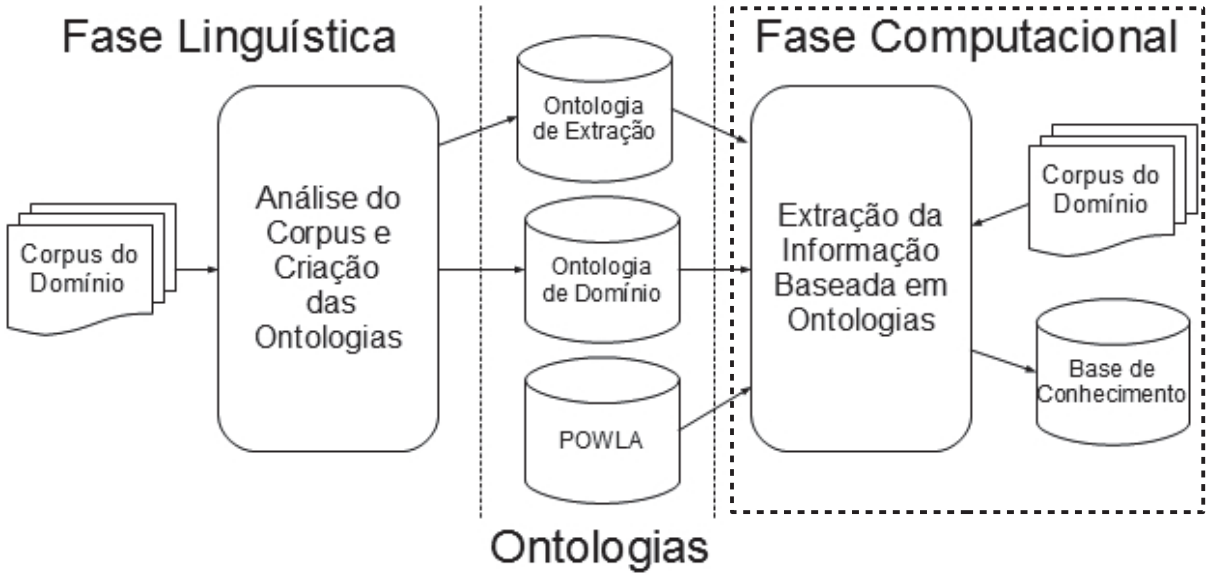
Relembrando o modelo geral, ilustrado na Figura 37, na Fase Computacional estão disponibilizadas três das quatro ontologias utilizadas no processo de EI: a ontologia de domínio *ODomJurBR,* a ontologia de extração *ODomJurBR-RAC* e a ontologia para modelagem de anotações linguísticas (POWLA).

As ontologias de domínio ODomJurBR e de extração ODomJurBR-RAC são obtidas ao final da Fase Linguística do estudo de caso, sendo o modelo de dados POWLA um arquivo OWL disponível na Internet[39].

Esta seção demarca o início da Fase Computacional do modelo, etapa em que são implementados os procedimentos necessários para realização do processo de EI. Apresenta-se

[39] Disponível em: <http://purl.org/powla/powla.owl>. Acesso em: 11 jul. 2013.

Figura 37: Visão geral do modelo com destaque para a Fase Computacional.



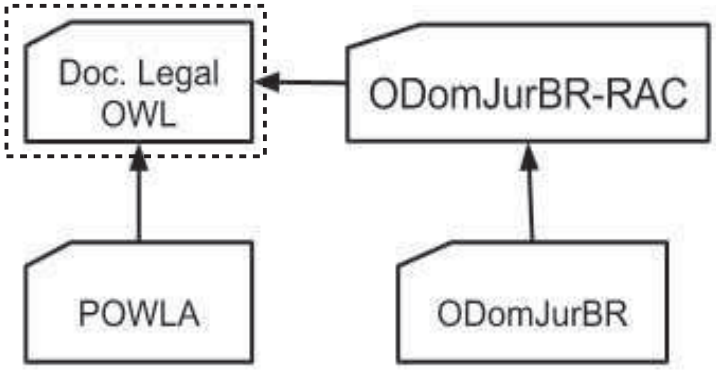
Fonte: elaborado pelo autor.

a partir de agora como foram implementado os processos, comentando, quando relevante, as dificuldades e respectivas soluções encontradas no transcorrer do experimento.

Conforme definido no capítulo anterior, o modelo sugerido neste trabalho prevê uma quarta ontologia: a representação em OWL do documento a ser analisado (destacada pelo retângulo tracejado na Figura 38). No experimento aqui sendo descrito, o documento é um Acórdão jurídico, o qual é linguisticamente anotado pelo parser Palavras e representado no modelo de dados POWLA.

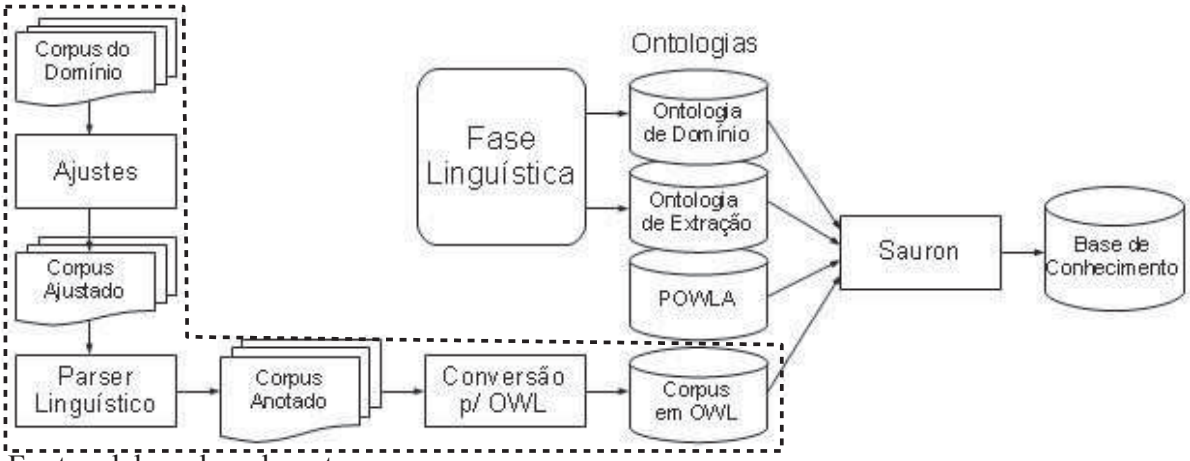
A geração da representação em OWL do documento linguisticamente anotado é realizada de forma totalmente automatizada, através da submissão de cada um dos documentos do corpus a uma sequência de processos que convertem os arquivos desde o formato texto puro até a sua representação em OWL.

Figura 38: Ontologias do modelo proposto.



Fonte: elaborado pelo autor.

Figura 39: Processos de conversão do corpus para OWL da Fase Computacional.



Fonte: elaborado pelo autor.

Como pode ser visto, a Figura 39 apresenta novamente a visão geral do modelo vista no capítulo anterior, desta vez destacando na área tracejada os processos diretamente envolvidos na conversão do corpus do domínio para o formato OWL.

Neste estudo de caso, toda a sequência de processos necessários para a conversão dos documentos em OWL foi implementada de forma a não ser necessária a intervenção humana, utilizando uma composição de aplicações computacionais já disponíveis e, quando necessário, desenvolvendo-se os sistemas necessários para realizar a conversão do documento ao formato OWL.

Nas próximas subseções serão detalhados cada um dos processos de conversão implementados no decorrer do estudo de caso, relatando-se também os problemas enfrentados e as respectivas soluções de contorno adotadas.

5.3.1 Ajustes do Texto dos Acórdãos

No decorrer do experimento, foram detectados alguns problemas ao submeter-se o arquivo-texto ao parser linguístico Palavras, resultando em erros de análise linguística ou geração de arquivos corrompidos, tornando necessário realizar-se alguns ajustes no arquivo antes da sua submissão ao parser (área destacada da Figura 40).

Figura 40: O processo de conversão para OWL, com destaque para o Pré-processamento.

Fonte: elaborado pelo autor.

Tabela 4: Exemplos de ajustes realizadas na fase de pré-processamento.

Frase original	Frase alterada
1) Participaram do julgamento, além do signatário, os eminentes Senhores **Des.** Newton Brasil de Leão (Presidente e Revisor) e **Des**. Manuel José Martinez Lucas.	Participaram do julgamento, além do signatário, os eminentes Senhores **Desembargador** Newton Brasil de Leão (Presidente e Revisor) e **Desembargador** Manuel José Martinez Lucas.
2) Esta Primeira Câmara Criminal julgou o Recurso em Sentido Estrito nº 70037693488 e, à unanimidade, negou provimento (**fls.** 428/435).	Esta Primeira Câmara Criminal julgou o Recurso em Sentido Estrito nº 70037693488 e, à unanimidade, negou provimento (**folhas** 428/435).
3) MP – Ele apresentou a vocês alguma arma que o Marciano portasse? T – Sim, aí nos tinha**...** aí veio a**...** a gente**...** foi uma pistola inox 380 quadrada ela era**...** raspada era.	MP – Ele apresentou a vocês alguma arma que o Marciano portasse? T – Sim, aí nos tinha**.** aí veio a**.** a gente**.** foi uma pistola inox 380 quadrada ela era**.** raspada era.
4) Assim, a pena privativa de liberdade fica definitivamente fixada em 09 (nove) meses e 23 **(vinte e três)** dias de reclusão.	4) Assim, a pena privativa de liberdade fica definitivamente fixada em 09 (nove) meses e 23 dias de reclusão.

A Tabela 4 apresenta exemplos de trechos retirados do corpus utilizado no estudo de caso que ilustram os ajustes efetuados. Importante ressaltar que são realizadas somente alterações superficiais no texto e que não comprometem o significado original da frase.

Os exemplos 1 e 2 da tabela apresentam dois casos de abreviaturas que são expandidas por causarem erro no processo de segmentação do texto que o *parser Palavras* realiza. Devido a um erro de interpretação do *Palavras*, este divide a frase ao encontrar o ponto final de algumas abreviaturas comumente usadas em Acórdãos.

O caso 3 está também relacionado ao problema de segmentação de frases, pois o *parser Palavras* segmenta as reticências em 3 frases, sendo 2 frases "vazias". Muito embora as reticências tenham como função linguística a separação de frases, as frases vazias causavam a geração de arquivos corrompidos, impossibilitando o andamento da conversão. Assim, a substituição das reticências pelo ponto final tornou-se uma ação necessária.

O caso 4 apresenta uma situação muito específica: após a constatação de diversos arquivos TIGER-XML com a sua estrutura XML corrompida, verificou-se que o padrão linguístico comumente utilizado para a escrita de numerais por extenso causavam um problema no *script* de conversão para TIGER-XML, gerando um arquivo XML defeituoso. A solução encontrada para este problema foi remover a escrita por extenso e os respectivos parênteses.

Estes e mais alguns outros ajustes são realizados de forma automatizada, através da aplicação de expressões regulares (FRIEDL, 2006) para a localização dos padrões de caracteres e respectiva substituição ou remoção.As expressões regulares são sequencialmente listadas em um arquivo, sendo aplicadas sobre o texto pelo comando *sed* [40]. O Quadro 9 apresenta, a título de exemplo, um trecho do arquivo utilizado para a realização dos ajustes nos documentos deste estudo de caso.

[40] <http://www.gnu.org/software/sed/manual/html_node/sed-Programs.html>. Acesso em: 30 Oct. 2013.

Quadro 9: Exemplos de Expressões Regulares utilizadas no pré-processamento.

```
#Remoção incondicional de aspas duplas e simples (costumam causar problemas)
s/'|`|"|“|”//g

#ABREVIATURAS
#Desembargador
s/(\<)(Des|DES)\./\1Desembargador/g
#Folhas
s/(\<)(fls|Fls|FLS)\./\1folhas/g
#Arts. => Artigos
s/(\<)([Aa])rts\./\1\2rtigos/g
s/(\<)ART(S)?\./\1ARTIGO\2/g
# Págs. => Páginas
s/(\<)(págs|Págs|PÁGS)\./\1páginas/g
# Expressões latinas abreviadas
s/(\<)op\. cit\./\1obra citada/g

#O Palavras tem problemas com reticências ("...") - substituindo por "."
s/\.\.\././g

# O problema centra-se o "e"
s/(\<)([0-9]+) \([ ]*[^ ]+ e [^ ]+\)/\1\2/g
```

Realizados os ajustes, o arquivo está pronto para a próxima etapa do processo de conversão: a submissão do documento ao *parser Palavras* para a geração do arquivo linguisticamente anotado.

5.3.2 Submissão ao Parser Linguístico

Após a realização dos ajustes, o documento é submetido ao analisador Palavras (BICK, 2000), o qual produz como saída o arquivo linguisticamente anotado (área em destaque da Figura 41). O arquivo linguisticamente anotado é gerado em um formato denominado de *árvore deitada*.

Conforme já citado anteriormente, não existe ainda um conversor do padrão de anotação linguística *árvore deitada* para o formato POWLA/OWL, sendo então necessário primeiro converter-se o arquivo linguisticamente anotado para um formato intermediário. A utilização do TIGER-XML para a representação intermediária foi então adotada como solução, utilizando-se o *script visl2tiger.pl* para a conversão do arquivo.

Uma vez convertido o arquivo para TIGER-XML, visando-se a automação plena do processo de conversão, buscou-se uma ferramenta computacional que verificasse a

Figura 41: Submissão do *Corpus Ajustado* ao parser *Palavras*.

Fonte: elaborado pelo autor.

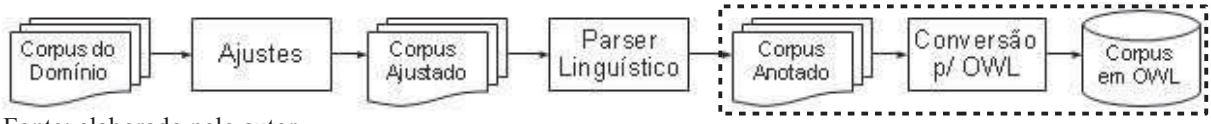
integridade do arquivo XML gerado. A validação do arquivo é realizada pela sua submissão ao comando *xmllint*[41], utilizado para a validação da integridade sintática de arquivos XML.

Esta etapa do processo se encerra com a disponibilização de um arquivo com a representação das informações linguísticas dos documentos jurídicos no formato TIGER-XML, sendo ainda necessário converter-se estas informações para o formato OWL, último processo ao qual o documento original é submetido.

5.3.3 Representação do Documento em OWL

A conversão do arquivo TIGER-XML é o último procedimento necessário para a almejada representação do Acórdão jurídico linguisticamente anotado em OWL, conforme pode ser visto em destaque na Figura 42.

Figura 42: Conversão do *Corpus Anotado* para OWL.



Fonte: elaborado pelo autor.

A conversão para OWL é realizada utilizando-se o modelo de dados POWLA. Conforme já comentado anteriormente, o *script* XSLT *tiger2owl.xsl* tem como função a conversão de arquivos no formato TIGER-XML para o formato OWL. Contudo, devido a incompatibilidades entre o arquivo gerado pelo *script visl2tiger.pl* e o padrão esperado pelo *script tiger2owl.xsl*, não foi possível utilizá-lo diretamente, sendo necessário realizar algumas alterações no seu código, originando-se daí um novo *script* chamado de *palavras2owl.xsl*.

Para a execução do *script palavras2owl.xsl* escolheu-se o comando *xsltproc* [42], uma ferramenta de linha de comando para aplicação de *scripts* XSLT a documentos XML e que utiliza a biblioteca *libxslt*[43].

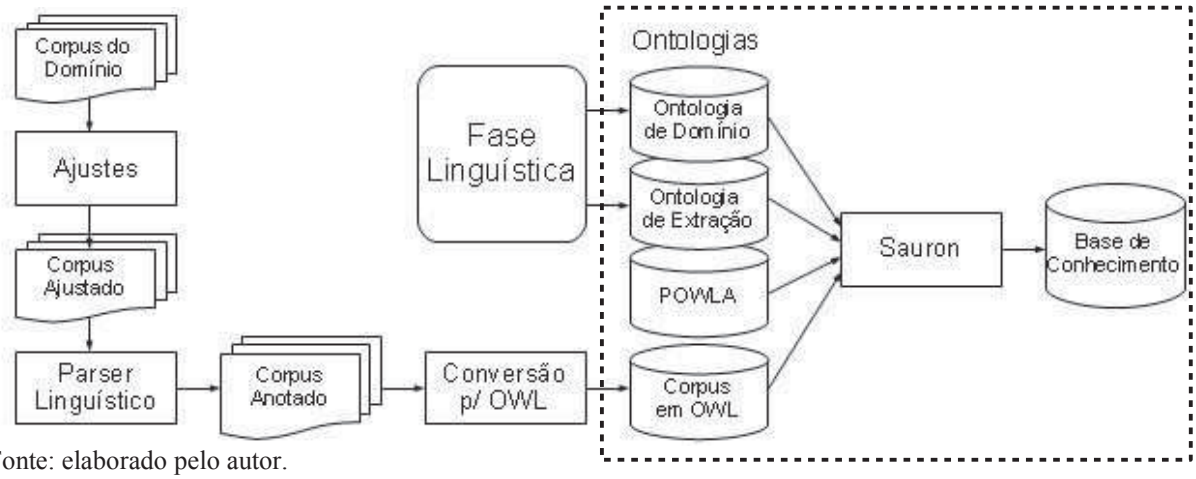
Representadas as anotações linguísticas no formato OWL, tem-se finalizado os processos de conversões do documento, encontrando-se prontas as quatro ontologias necessárias para a realização do processo de extração das informações através do sistema SAURON.

5.4 Extração das Informações via Sistema SAURON

Gerada a representação OWL do texto e das anotações linguísticas do Acórdão, estão prontas as quatro ontologias necessárias para a realização do processo de EI, o qual será efetivado pelo sistema SAURON, como pode ser visto em destaque na Figura 43.

[41] <http://xmlsoft.org/xmllint.html>. Acesso em: 11 jul. 2013.
[42] <http://xmlsoft.org/XSLT/xsltproc2.html>. Acesso em: 11 jul. 2013.
[43] <http://xmlsoft.org/XSLT>. Acesso em: 11 jul. 2013.

Figura 43: O sistema SAURON e as ontologias da Fase Computacional.



Fonte: elaborado pelo autor.

No estudo de caso desenvolvido, optou-se pela seguinte abordagem: o sistema SAURON implementaria exclusivamente o processo de extração das informações a partir dos documentos no formato OWL. A conversão dos Acórdãos para o formato OWL não faz parte do escopo do sistema SAURON, sendo implementado por um *script* desenvolvido em Bash [44], que realiza a sequência de processos necessários para a conversão e validação dos documentos desde o formato texto puro, até a sua representação em OWL.

Com base nestas premissas, a interface gráfica do sistema SAURON foi implementada de forma a permitir que sejam selecionados o conjunto de arquivos OWL a serem processados. O processamento dos arquivos acontece pela repetição da seguinte sequência de procedimentos para cada OWL selecionado:

a. carga das ontologias ODomJurBR, ODomJurBR-RAC, POWLA e do Acórdão representado em OWL;
b. combinação por mesclagem das quatro ontologias, agrupando todos os axiomas e regras em uma única ontologia;
c. disparo do reasoner para a realização das inferências lógicas e consequente extração das informações;
d. realização de uma consulta SPARQL para a busca de todos os indivíduos inferidos da classe Eventos
e. geração da Base de Conhecimento, que compõe-se de um arquivo-texto contendo os eventos localizados.

Repetidos os procedimentos acima para cada arquivo OWL, resulta um conjunto de arquivos textos contendo informações para a localização das referências aos *Eventos* jurídicos da ontologia de domínio ODomJurBR nos Acórdãos submetidos ao processo de EI.

[44] <http://www.gnu.org/software/bash/manual/bashref.html>. Acesso em: 30 out. 2013

Quadro 10: Conteúdo de um arquivo da Base de Conhecimento.

Evento: **Denúncia**

Nodos : s3_501 – s3_1 s3_2 s3_3 s3_4 s3_5 s3_6 s3_7 s3_8 s3_9 s3_10 s3_11 s3_12 s3_13 s3_14 s3_15 s3_16 s3_17 s3_18 s3_19 s3_20 s3_21 s3_22 s3_23 s3_24 s3_25 s3_26 s3_27 s3_28 s3_29 s3_30 s3_31 s3_32

Sintagma: O Ministério_Público **denunciou** XXXX XXXXX XXXXXXXXXXXX XXXXXXXX por incurso em as sanções de o art. 184 , § 2º , de o Código_Penal , por a prática de o seguinte fato delituoso :

O apresenta um trecho típico de um arquivo-texto da Base de Conhecimento. O trecho refere-se a um Acórdão no qual localizou-se uma referência ao evento *Denúncia* no nodo *Nonterminal* identificado pelo código *s3_501*. Os códigos *s3_1* a *s3_32* referem-se aos nodos *Terminal* que compõem o sintagma *s3_501*.

O conteúdo do arquivo-texto referente a Base de Conhecimento é composto por três linhas de informação para cada evento localizado: a primeira linha apresenta o nome do evento jurídico localizado, a segunda linha apresenta os códigos identificadores dos nodos que contém a referência e por fim apresenta-se o sintagma em si.

Com objetivo de validar e verificar o desempenho da abordagem proposta neste trabalho, realizou-se um primeiro experimento, aplicando-se a metodologia a um conjunto de Acórdãos do corpus previamente coletado. Os resultados obtidos neste primeiro experimento são relatados da próxima seção.

5.5 Realização e Resultados do Primeiro Experimento

O primeiro experimento refere-se a aplicação da abordagem de EI proposta neste trabalho a um corpus de 200 Acórdãos. O corpus utilizado para a realização do experimento compõe-se de um novo conjunto de documentos, distinto do utilizado na Fase Linguística.

O corpus utilizado neste documento totalizou 39.895 sentenças e 618.892 palavras, cobrindo decisões proferidas por 19 Desembargadores do Tribunal. Conforme descrito na seção anterior, os 200 Acórdãos no formato texto foram convertidos para OWL, gerando os respectivos arquivos linguisticamente anotados e convertido ao modelo de dados POWLA.

Tabela 5: Quantidade de eventos identificados no primeiro experimento.

Eventos	# Referências Identificadas
Absolvição	28
Condenação	97
Denúncia	98
Interrogatório	6
Total	229

Estes 200 arquivos OWL foram selecionados através da interface gráfica do sistema SAURON para serem submetidos sequencialmente ao processo de EI, gerando a respectiva Base de Conhecimento no formato texto.

A Base de Conhecimento gerada neste experimento identificou eventos jurídicos da ontologia de domínio ODomJurBR em 136 Acórdãos, sendo localizadas referências em 229 sintagmas, distribuídas conforme apresentado na Tabela 5.

Para a verificação do desempenho do processo de EI implementado, utilizou-se como procedimento a verificação manual dos 200 Acórdãos, realizada por dois pesquisadores do grupo de pesquisa computacional, os quais fizeram um levantamento da quantidade de acertos e erros da abordagem, classificando-os em um dos seguintes tipos:

Verdadeiro Positivo (VP): é contado um VP quando o sistema inferencial identifica a ocorrência de um evento jurídico e confirma-se na conferência manual que o sintagma contém realmente uma ocorrência do evento (acerto);

- Verdadeiro Negativo (VN): o sintagma contém uma palavra cujo o radical poderia levar as RACs a erroneamente reconhecer ali um evento, o que efetivamente não aconteceu (acerto);
- Falso Positivo (FP): uma RAC identificou um evento em um sintagma e na conferência manual verificou-se sua inocorrência (erro);
- Falso Negativo (FN): é a ocorrência de um sintagma que contenha uma referência a um evento jurídico mas que as RACs não conseguiram identificar (erro).

O procedimento de contabilização dos *Positivos* seguiu a seguinte metodologia: a partir dos arquivos da Base de Conhecimento, buscou-se localizar o sintagma no texto original do Acórdão para verificação da existência ou não da referência ao evento, contabilizando-se assim acertos (VP) ou erros (FP) das RACs.

O procedimento de buscar-se no texto original do Acórdão o sintagma tinha dois objetivos principais: primeiro, verificar se a consulta SPARQL utilizada para a geração da Base de Conhecimento produzia a remontagem correta da sentença e, segundo, em determinadas situações o sintagma não continha palavras suficientes para identificar-se com certeza o contexto da frase, sendo necessário então analisar todo o parágrafo no qual o sintagma estava contido.

No Quadro 11 apresenta-se um exemplo de uma situação em que as informações apresentadas na Base de Conhecimento são insuficientes para a confirmação do acerto ou erro da RAC quanto à presença do evento jurídico *Absolvição*.

No arquivo da Base de Conhecimento, o sintagma "*absolvendo os acusados*" não apresenta o agente do verbo *absolver*, sendo então impossível verificar se houve acerto (VP) ou erro (FP) da RAC. No entanto, ao analisar-se o sintagma no contexto do parágrafo, verificamos que a absolvição é consequência de uma sentença judicial, confirmando-se assim um acerto do tipo VP.

Quadro 11: Sintagma com palavras insuficientes para análise do contexto.

Base de Conhecimento:

Evento: Absolvição

Nodos : s7_513 – s7_28 s7_29 s7_30

Sintagma: ***absolvendo os acusados***

Sintagma no Acórdão:

A denúncia foi recebida em 14.05.2007 (fl. 70), sendo que, após regular trâmite processual, adveio sentença que julgou improcedente a ação penal, ***absolvendo os acusados****.*

A contabilização dos *Negativos*, no entanto, seguiu um procedimento diferente. Para a classificação de *Verdadeiro* ou *Falso* dos casos *Negativos* adotou-se a seguinte estratégia: utilizando-se o *script* PERL *txt2pdf.pl* [45], foram convertidos todos os 200 Acórdãos para o formato PDF, destacando-se as sequências de caracteres que casassem com o radical das palavras utilizadas para representar os eventos na ODomJurBR.

Em linguística, o radical de uma palavra é o seu elemento estrutural básico, que expressa significado e não apresenta variação. Por exemplo, para o termo *Absolvição*, que é utilizado para a representação do evento jurídico de mesmo nome na ODomJurBR, tem-se como radical a sequência de caracteres "*absolv*".

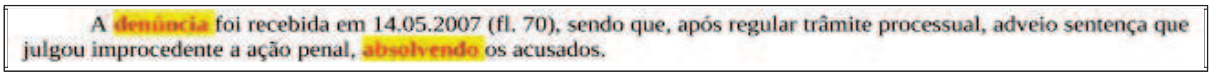
O *script txt2pdf.pl* permite que sejam gerados arquivos PDF com uma coloração específica para uma sequência de caracteres, facilitando assim a localização visual das palavras que contém o radical dos eventos da ODomJurBR no decorrer da análise manual dos Acórdãos.

A contabilização dos *Negativos* dá-se então pela busca dos radicais dos eventos nos Acórdãos em PDF, classificando-se como *Falso Negativo* a ocorrência de um evento existente no Acórdão e cuja a presença não foi identificada na Base de Conhecimento (erro) e de *Verdadeiro Negativo* quando ocorre a presença do radical em uma sentença em que não ocorre o evento e na Base de Conhecimento não existe apontamento para esta sentença (acerto).

A análise de um caso prático ajudará a compreender a metodologia adotada para a contagem de erros e acertos da abordagem. O trecho do Acórdão convertido para PDF visto na Figura 44 ilustra a ocorrência de duas palavras cujo o radical remete, pela sua composição léxica, aos eventos da ODomJurBR e poderiam levar as RACs a concluir a presença de dois eventos jurídicos.

No caso do sintagma que contém a palavra "*denúncia*" observa-se a ocorrência do evento *recebimento de denúncia*, evento não representado na ODomJurBR. Há uma diferença sutil entre os eventos de recebimento e de apresentação da denúncia. Na ontologia

Figura 44: Trecho do PDF gerado a partir da aplicação do *script txt2pdf.pl*.


A denúncia foi recebida em 14.05.2007 (fl. 70), sendo que, após regular trâmite processual, adveio sentença que julgou improcedente a ação penal, absolvendo os acusados.

Fonte: elaborado pelo autor.

[45] Disponível em: <http://www.sanface.com/txt2pdf.html>. Acesso em: 30 out. 2013.

ODomJurBR representa-se apenas o conceito de apresentação da *Denúncia*, sendo então elaboradas RACs para o reconhecimento somente deste evento.

Se houver na Base de Conhecimento, após a execução do sistema SAURON, o apontamento de *Denúncia* para o sintagma "*A denúncia foi recebida em 14.05.2007*", será classificado como erro do tipo *FP*. Por outro lado, se na Base de Conhecimento não houver apontamento para o evento *Denúncia* nesta frase, então será contado um acerto do tipo *VN*.

Em relação ao sintagma "*absolvendo os acusados*", ali se encontra presente a ação de absolvição jurídica que se quer representar com a classe *Absolvição* da ontologia de domínio, pois o agente do verbo *absolver* é o sintagma nominal "*a sentença*". Havendo a identificação do evento *Absolvição* na Base de Conhecimento, conta-se como um acerto do tipo VP. Em não havendo o reconhecimento do evento, conta-se então um erro do tipo FN.

A Tabela 6 apresenta a contagem de erros e acertos do experimento, bem como a quantidade de erros ocorridos por não haver regras linguísticas para identificação do padrão de extração (SR). A contagem SR refere-se a situações onde nota-se a ocorrência do evento que não foram identificados por não haver RACs para o padrão linguístico. Ou seja, são os padrões linguísticos não mapeados em regras.

Tabela 6: Contagens realizadas para o primeiro experimento.

	Absolvição	Condenação	Denúncia	Interrogatório	Total
VP	26	89	102	6	223
FP	1	2	0	0	3
VN	211	258	919	10	1398
FN	20	61	102	76	259
SR	20	61	102	76	259

Os dados gerados por este primeiro experimento foram registrados e efetuados os cálculos de *Precisão, Acurácia* e *Revocação* (ANDROUTSOPOULOS; MALAKASIOTIS, 2009). No experimento realizado, calcula-se a *Acurácia* analisando-se erros e acertos a partir da Base de Conhecimento e dos Acórdãos (fórmula 5.1). Considera-se a *Precisão* como sendo composta tão somente pela correção das informações apresentadas na Base de Conhecimento (fórmula 5.2).

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \tag{5.1}$$

$$Precisão = \frac{VP}{VP + FP} \tag{5.2}$$

Legenda: VP = Verdadeiro Positivo *VN = Verdadeiro Negativo*
FP = Falso Positivo *FN = Falso Negativo*

A pretendida abrangência de análise com a *Acurácia* (fórmula 5.1) decorre da contabilização dos acertos (VN) e erros (FN) tendo-se como base não somente a identificação dos conceitos do domínio realizadas pelas RACs (registrados na Base de Conhecimento), mas

também a capacidade de identificação de estruturas linguísticas (presentes nos Acórdãos) que poderiam levar as RACs a identificar erroneamente a presença de conceitos.

A *Precisão* (fórmula 5.2) apresenta uma análise totalmente focada nos resultados apresentados na Base de Conhecimento gerada e, por este enfoque, também é uma métrica a ser considerada no momento de analisar-se o desempenho do processo de EI.

Outra métrica de desempenho utilizada para avaliação do desempenho da abordagem é a *Revocação*, a qual calcula a proporção de identificações corretas em relação ao total geral de eventos existentes no corpus, resultando na fórmula 5.3 apresentada abaixo.

$$Revocação = \frac{VP}{VP + FN} \tag{5.3}$$

A Tabela 7 apresenta o desempenho obtido neste primeiro experimento considerando-se a *Acurácia*, a *Precisão* e a *Revocação*. São apresentadas a seguir algumas reflexões sobre os resultados obtidos.

Tabela 7: Desempenho do primeiro experimento.

Evento	Acurácia	Precisão	Revocação
Absolvição	0,92	0,96	0,57
Condenação	0,84	0,98	0,58
Denúncia	0,86	1,00	0,40
Interrogatório	0,17	1,00	0,07
Geral	0,86	0,99	0,46

Em primeiro lugar, comparando-se as métricas, verifica-se um melhor desempenho geral na *Precisão* do que na *Revocação*. Esta é uma característica inerente ao processo de EI baseado em regras linguísticas, pois estas permitem uma melhor avaliação do significado da frase, uma vez que se fundamenta em informações de vários níveis do processamento da linguagem natural para a avaliação da presença ou não do conceito.

A elaboração de regras linguísticas a partir de estudo do corpus busca exatamente abranger a maior quantidade possível de casos, mas sabe-se que, por maior que seja a quantidade de regras, nunca serão esgotadas todas as possibilidades de expressão dos conceitos. O objetivo é descrever-se regras linguísticas com níveis de expressividade e generalização que identifiquem a maior quantidade possível de casos.

Ainda que levados em conta as argumentações acima, verificou-se um desempenho muito aquém do aceitável para a *Revocação*. O índice de 7% alcançado para o evento *Interrogatório*, por exemplo, chamou a atenção, denotando a possibilidade de algum fator externo influenciar negativamente o desempenho.

A ideia de influências externas é reforçada pela diferença observada entre os desempenhos apresentados nos de cálculos da *Precisão* e *Acurácia*. Observa-se, novamente

para o evento *Interrogatório*, uma queda no desempenho extremamente acentuada para o índice de *Acurácia*.

Refletindo-se sobre as anomalias acima comentadas, conjectura-se como possível causa o fato de que para o desenvolvimento deste experimento não foi possível contar com a participação dos especialistas em linguística e de representação do conhecimento na etapa de análise do corpus, o que resultou em um número pequeno de Acórdãos analisados para a elaboração RACs. Além disto, deve-se também levar em conta a falta de experiência em análise de corpus do pesquisador, pois esta tarefa foi realizada por um especialista em computação.

Outro fator importante, na análise de corpus realizada na Fase Linguística deste estudo de caso foram utilizados somente 10 Acórdãos para a elaboração das regras da ontologia de extração. Esta amostragem, sabe-se agora, foi menor que a necessária para obter-se um número de frases que permitissem a geração de regras em quantidade suficiente para o reconhecimento das estruturas linguísticas em que os eventos costumeiramente ocorrem nos Acórdãos. O baixo número de Acórdãos analisados deveu-se principalmente aos seguintes fatores:

a. morosidade do processo de análise devido à falta de experiência e conhecimentos especializados em linguística e representação do conhecimento;

b. pouca disponibilidade de recursos humanos para a realização da análise, sendo esta realizada por somente um pesquisador do grupo computacional;

c. a falta de um sistema integrado para o desenvolvimento das regras linguísticas, sendo necessário criar, testar e analisar a abrangência das regras de forma totalmente manual, utilizando-se o editor de ontologias Protégé para esta tarefa.

A reduzida quantidade de 11 RACs formalizadas na ontologia ODomJurBR-RAC é o principal fator para o fraco desempenho para a $_{Acurácia}$. Esta conclusão é ratificada por duas constatações: a grande variação de amplitude do desempenho verificado nos resultados (indo de 0,17 a 0,92) e o excelente desempenho obtido ao considerar-se somente os resultados da Base de Conhecimento (*Precisão*).

O efeito da pequena amostragem de documentos analisada na Fase Linguística é evidenciado ao observar-se a acentuada discrepância nos desempenhos apontados no evento *Interrogatório*, saindo de irreais 100% de *Precisão* para irrisórios 17% de *Acurácia*. Em face a estas dicotomias observadas nos resultados obtidos para a *Precisão* e a *Acurácia* no evento *Interrogatório*, realizou-se um aprofundamento na análise das possíveis causas, encontrando-se outros fatores de influência, como por exemplo o reduzido número de ocorrências do evento *Interrogatório* no corpus utilizado para análise na Fase Linguística.

Visando verificar-se mais detalhadamente o efeito da quantidade de documentos analisados na Fase Linguística sobre o desempenho geral da abordagem, formulou-se um novo cálculo para o cálculo da *Precisão*, isolando-se o número de erros causados por não haver uma regra de extração para o padrão linguístico (fórmula 5.4).

$$Precisão_{SR} = \frac{VP + VN}{VP + VN + FP + FN - SR} \tag{5.4}$$

Considerando esta nova abordagem para o cálculo da Precisão, refizeram-se os cálculos para os eventos, resultando em uma nova escala de desempenho, a qual é apresentada na Tabela 8 juntamente aos índices anteriores de desempenho obtidos, a fim de facilitar-se a comparação entre as diferentes fórmulas utilizadas. Observa-se então uma melhora nos índices de *Precisão* ao retirar-se do cálculo o grau de generalização das regras, aproximando-se consideravelmente os resultados obtidos para a $Precisão_{SR}$ e a *Precisão*.

Tabela 8: Desempenhos obtidos no primeiro experimento.

Evento	Acurácia	Precisão	$Precisão_{SR}$	Revocação
Absolvição	0,92	0,96	0,98	0,57
Condenação	0,84	0,98	0,95	0,58
Denúncia	0,86	1,00	0,99	0,40
Interrogatório	0,17	1,00	1,00	0,07
Geral	0,86	0,99	1,00	0,46

Diante desta situação, a fim de comprovar-se a teoria de que o desempenho observado nas métricas de *Acurácia* e *Revocação* são realmente influenciados pela pouca quantidade de documentos analisados na Fase Linguística e não por algum outro fator, decidiu-se pela realização de um segundo experimento, o qual será a seguir relatado.

5.6 Preparação para o Segundo Experimento

Como visto na seção anterior, a fim de comprovar-se e relação entre o desempenho de EI pelo modelo proposto neste trabalho e o tamanho da amostragem utilizada na análise na Fase Linguística do estudo caso, decidiu-se pela realização de um segundo experimento.

Aproveitou-se também para a implementação de algumas otimizações operacionais na Fase Computacional, visando obter-se menores tempo de execução e consumo de memória no decorrer do processo de EI realizado pelo sistema SAURON, pois observou-se preocupante o tempo e a memória consumidos para o processamento dos 200 Acórdãos.

Inicia-se então esta seção apresentando as duas otimizações, uma implementada na conversão do Acórdão em OWL e outra na forma de realização do processo de extração executado pelo sistema SAURON.

5.6.1 Exclusão dos Nodos Insignificantes – ENI

A realização do primeiro experimento deixou evidente um altíssimo consumo de memória para o processamento dos Acórdãos no sistema SAURON. Dada a simplicidade de operações realizadas no código do sistema, conclui-se que o consumo de memória relacionava-se às operações inferenciais realizadas pelo *reasoner*.

Sendo a lógica inferencial o núcleo operacional do processo de EI, bem como o principal diferencial proposto neste trabalho, realizaram-se análises para diminuir-se o consumo excessivo de memória.

Analisando-se o processo de conversão dos Acórdãos, verificou-se que a representação no modelo de dados POWLA resultava em arquivos OWL extremamente extensos. Um

Acórdão no formato texto, por exemplo, com tamanho de 15 Kbytes resulta, após a conversão para OWL, em um arquivo com 10,5 Mbytes, representando um crescimento de 700 vezes o tamanho original.

O tamanho médio dos arquivos no formato texto dos Acórdãos é 21 Kbytes, prevendo-se então uma média de 14,7 Mbytes de tamanho para os arquivos OWL. Considerando-se o modo de funcionamento da lógica inferencial, verifica-se que o tamanho dos arquivos com a representação em OWL dos Acórdãos é uma questão importante a ser resolvida, pois tem impacto não somente na quantidade de memória consumida ao disparar o reasoner, mas também no tempo de processamento necessário para a verificação das regras de extração.

A solução encontrada para a redução do tamanho dos arquivos OWL foi remover todas as informações que não fossem significativas para o processo de extração, partindo-se de uma premissa simples: são significantes os nodos Terminais e as respectivas árvores sintáticas que estão efetivamente referenciados nas RACs.

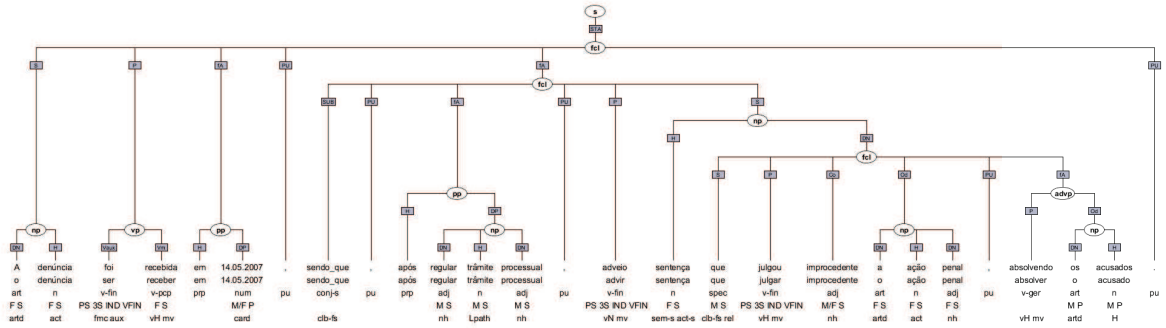
Os demais nodos, por analogia, insignificantes para o processo de extração, podem ser removidos da representação em OWL. As RACs contém regras que verificam se determinadas características linguísticas estão presentes na frase para então concluir se há ou não referência a uma informação de interesse. A caracterização linguística faz-se pela análise de alguns poucos elementos do tipo *Terminal* (ou seja, os termos das frases) e as relações sintáticas entre estes elementos.

Com base nesta constatação, desenvolveu-se um sistema em Java que implementa um algoritmo para Exclusão de Nodos Insignificantes (ENI), o qual carrega a ontologia de extração e identifica todos os nodos do tipo *Terminal* que são utilizados nas regras linguísticas de extração, criando uma lista dos nodos *significantes*.

Criada a lista dos nodos significantes, o sistema carrega o arquivo TIGER-XML e verifica cada nodo *Terminal*, removendo todos aqueles que não constem na lista de nodos *significantes*. Após a remoção dos nodos insignificantes, diversas árvores de sintaxe ficam vazias, ou seja, sem nodos folhas do tipo *Terminal*, sendo então necessário percorrer

Figura 45: Efeito do algoritmo ENI sobre uma frase.

A denúncia **foi** recebida em 14.05.2007 (fl. 70), sendo que, após regular trâmite processual, adveio **sentença** que **julgou** improcedente a ação penal, **absolvendo** os **acusados**.



Fonte: elaborado pelo autor.

novamente o arquivo TIGER-XML para a remoção dos nodos *Nonterminal* participantes de árvores vazias.

Na Figura 45 pode-se ver uma frase retirada de um dos Acórdãos do corpus e a sua respectiva árvore sintática e na Figura 46 são mostrados os nodos significantes que permaneceram após a submissão do arquivo TIGER-XML ao algoritmo ENI.

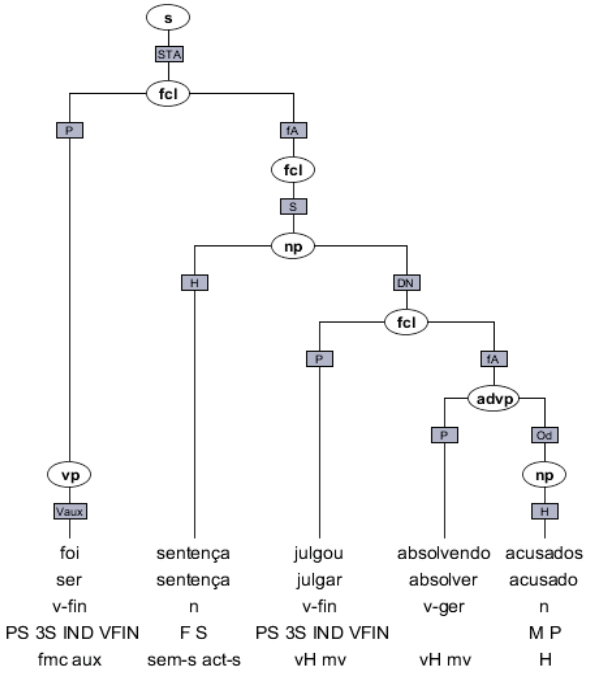
O processo de redução é extremamente efetivo, diminuindo os arquivos TIGER-XML para 25% do tamanho original em média. O reflexo da aplicação do algoritmo ENI aos arquivos TIGER-XML na conversão para o formato OWL foram ainda mais significativos, reduzindo o espaço ocupado em disco dos 200 arquivos OWL de 500 para 45 Mbytes, ou seja, uma redução de 91% em média.

O único efeito colateral da adoção do algoritmo ENI é que as regras de extração devem sempre identificar a forma canônica ("*lemma*") dos nodos *Terminal*, pois este é hoje o único atributo avaliado pelo algoritmo para a determinação da *significância* do nodo.

No entanto, esta restrição tem pouquíssimo efeito sobre a elaboração das regras, pois a identificação da palavra pelo seu "*lemma*" é a forma mais comum de verificação do seu significado.

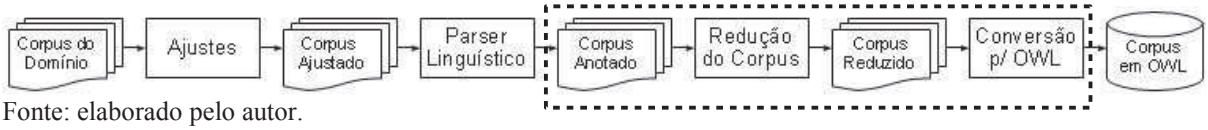
A alta taxa de redução do tamanho dos arquivos alcançada pela adoção do algoritmo *ENI* teve como efeito também a redução do tempo de execução do processamento dos 200 Acórdãos no sistema SAURON, que passou de 30,3 para 9,5 horas.

Figura 46: Árvore após aplicação do algoritmo ENI.



Fonte: elaborado pelo autor.

Figura 47: Fluxo de processos da conversão para OWL com a inclusão do Algoritmo ENI.



Fonte: elaborado pelo autor.

O algoritmo foi inserido na sequência de conversão dos Acórdãos, imediatamente após a geração do arquivo TIGER-XML, alterando o fluxo geral de conversão dos Acórdãos em OWL, incluindo definitivamente esta otimização operacional na Fase Computacional, denominando-a genericamente de *Redução do Corpus*, conforme destacado pela linha pontilhada na Figura 47.

A inserção do algoritmo ENI no fluxo de processos trouxe um ganho expressivo de desempenho , conforme já citado anteriormente. Contudo, visando-se a aplicação da abordagem a situações do mundo real, projetou-se ainda uma remodelagem do sistema SAURON, a qual será detalhada na próxima seção.

5.6.2 Sistema SAURON Sentença a Sentença

Conforme já descrito anteriormente, o processo efetivo de EI é gerenciado pelo sistema SAURON, que faz a carga das quatro ontologias do modelo e dispara o reasoner para a aplicação das RACs às frases do Acórdão.

Mesmo com a drástica redução do tamanho dos arquivos OWL proporcionada pela aplicação do algoritmo ENI, em situações onde houver textos muito extensos ainda haverá um alto consumo de memória e tempo de processamento para a realização da EI.

Além disto, o conjunto de regras formalizadas na ontologia de extração tende naturalmente a ser bastante extenso, pois para que se alcance bom desempenho na identificação das informações de interesse necessita-se um conjunto de regras que mapeie os diversos padrões linguísticos utilizados para o reconhecimento dos conceitos da ontologia de domínio.

Um conjunto grande de regras demandará um maior esforço computacional para a realização do processo de EI e mais espaço de memória ocupado para a realização das inferências lógicas. Com vistas à aplicação da metodologia aqui proposta em situações do mundo real, reprojetou-se o funcionamento do sistema SAURON, de modo que seja possível manipular tanto documentos muito extensos quanto ontologias de extração com grande quantidade de regras.

Desenvolveu-se então, utilizando exatamente os mesmos recursos (linguagem Java e OWL API), uma nova versão do sistema SAURON. Nesta versão, o processo de EI é implementado de uma forma levemente diferenciada.

O arquivo que contem a representação em OWL do Acórdão linguisticamente anotado é carregado em memória, igualmente ao procedimento anterior. No entanto, em vez de mesclar o documento inteiro com as demais ontologias do modelo, utilizou-se uma abordagem diferente: foi desenvolvido um algoritmo que percorre o documento sentença a sentença, transformando cada sentença em uma ontologia e então realizando a mesclagem desta com as demais ontologias do modelo.

Do ponto de vista do processo de análise das sentenças realizado pelas RACs, não há nenhum impacto, pois atualmente as RACs buscam identificar os elementos de interesse (conceitos da ontologia de domínio) considerando as informações linguísticas da própria frase.

Desta forma implementado, o esforço computacional para a aplicação do processo de EI não depende mais do tamanho total do documento sendo analisado, mas sim do tamanho das frases que compõem o documento.

Esta nova abordagem de processamento dos arquivos OWL foi implementada, testada e aprovada, dando origem a uma nova versão do sistema, batizada de SAURON SaS (sentença a sentença).

Aproveitou-se também para, nesta nova versão, alterar-se o formato da Base de Conhecimento gerada, sendo agora produzido, além do arquivo no formato texto, também um arquivo OWL como saída.

A Base de Conhecimento em OWL gerada pelo SAURON SaS contém somente os indivíduos do documento que tem relações com a ontologia de domínio. A partir das informações armazenadas na Base de Conhecimento é possível identificar os documentos e as sentenças que contém referência aos conceitos do domínio.

Importante salientar-se que as alterações realizadas não alteram a proposta de lógica unificada do processo de EI proposto neste trabalho, pois embora o sistema SAURON SaS tenha uma maior interação com os documentos representados em OWL, observa-se que toda a lógica de extração segue, tal qual na versão anterior, totalmente representada na ontologia de extração.

Combinadas as duas novas funcionalidades, algoritmo ENI e SAURON SaS, obteve-se uma solução de EI com excelentes perspectivas de aplicação a demandas reais. O tempo consumido para processar os 200 Acórdãos no primeiro experimento foram de aproximadamente 33,5 horas. Para a realização do segundo experimento, sobre o mesmo número de Acórdãos, necessitou de aproximadamente 1 hora de processamento. Importante frisar-se que ambos os experimentos foram realizados no mesmo servidor sob exatamente as mesmas condições.

Nas próximas seções deste capítulo relata-se os resultados obtidos neste segundo experimento realizado.

5.7 Realização e Resultados do Segundo Experimento

O primeiro experimento tinha como objetivo a comprovação da viabilidade da abordagem e verificação do desempenho frente ao corpus jurídico de Acórdãos. Confirmada a viabilidade, observou-se um desempenho aquém do desejado, principalmente em relação ao quesito *Revocação*.

A fim de comprovar-se a teoria de que o desempenho obtido no primeiro experimento foi resultado da quantidade insuficiente de regras de extração, consequência da reduzida quantidade de Acórdãos analisados na Fase Linguística, planejou-se um novo experimento, cuja metodologia adotada tem a seguinte linha de ação:

1. Selecionar os eventos jurídicos do primeiro experimento com o pior e o melhor desempenho no quesito Acurácia, ou seja, respectivamente os eventos Interrogatório e Absolvição;
2. Analisar todas as frases do corpus que fizerem menção aos dois eventos selecionados pelo procedimento acima, com vistas a identificar as características linguísticas destas sentenças para a elaboração de novas RACs;
3. Submeter novamente o corpus de 200 Acórdãos ao SAURON SaS para a extração das informações;
4. Realizar a contagem de erros e acertos para a realização do cálculo das métricas Acurácia e Revocação.

Antes de apresentarmos os números relativos ao desempenho obtido neste segundo experimento, é imprescindível explicitar-se que não se tem como objetivo a busca de melhores resultados, pois os desempenhos serão obrigatoriamente melhores do que no primeiro experimento, uma vez que 100% do corpus será analisado para a elaboração das regras linguísticas de extração deste segundo experimento.

Tem-se por objetivo agora validar a teoria de que uma análise de corpus realizada com uma amostragem adequada de documentos, com consequente maior quantidade de regras linguísticas, resulta em melhor desempenho da abordagem proposta neste trabalho. Caso não se observe uma melhora do desempenho neste segundo experimento, evidencia-se então que a abordagem de EI tem sua capacidade de extração limitada por outros fatores que não a quantidade e qualidade das regras de extração.

Feito este importante esclarecimento, inicia-se o relato dos resultados deste segundo experimento. A primeira consequência da análise dos 200 documentos é o expressivo crescimento do número de regras, que passou de 11 para 23 RACs linguísticas de extração [46]. Conforme procedimento realizado para o relato do primeiro experimento, apresenta-se na Tabela 9 a contagem de erros e acertos na localização dos eventos para o segundo experimento.

Tabela 9: Contagens realizadas para o segundo experimento.

Acertos/Erros	Absolvição	Interrogatório	Total
VP	32	65	97
FP	3	1	4
VN	208	7	215
FN	13	19	32

[46] As regras de extração utilizadas no segundo experimento estão disponíveis no Apêndice B.

Na Tabela 10 são apresentados para fins de comparação os desempenhos obtidos nos experimentos 1 e 2 quanto às métricas *Acurácia* e *Revocação*. Observa-se uma melhora considerável nos índices gerais, devido principalmente ao ganho de desempenho nas métricas relativas ao evento *Interrogatório*.

Tabela 10: Desempenhos obtidos no segundo experimento.

Evento	$Acurácia_1$	$Acurácia_2$	$Revocação_1$	$Revocação_2$
Absolvição	0,92	0,94	0,57	0,71
Interrogatório	0,17	0,78	0,07	0,77
Geral	0,72	0,90	0,25	0,75

Verifica-se um nos resultados apresentados um desempenho adequado para a *Acurácia*, indicando uma abordagem promissora, embora se observe ainda em relação à *Revocação* um espaço para melhorias. Principalmente, tem-se comprovada a teoria de que os baixos desempenhos do primeiro experimento tem como principal causa a quantidade de regras linguísticas elaboradas, causada pela pequena quantidade de documentos analisados na Fase Linguística.

Muito embora se tenha reduzido neste segundo experimento a quantidade de eventos a serem observados na análise de corpus, e apesar da maior experiência em relação ao processo de análise do corpus e elaboração das RACs, ainda assim se verifica uma grande morosidade para a verificação dos resultados obtidos pela inserção ou alteração das regras linguísticas, consumindo-se semanas de trabalho para a elaboração do novo conjunto de RACs.

Ficou evidente no transcorrer deste segundo experimento a necessidade de um sistema que proporcione um ambiente integrado para o desenvolvimento de RACs, o qual viabilize a criação de mais e melhores regras de extração em menos tempo. O desenvolvimento de tal sistema é um dos assuntos vistos no próximo capítulo.

6 CONCLUSÃO

Em relação à questão de pesquisa, formulada no capítulo 1 e reapresentada no Quadro 12, com base nas conclusões obtidas a partir dos experimentos realizados neste trabalho, mostrou-se evidenciado que o uso dos mecanismos de inferência da OWL em conjunto as ontologias de domínio e de extração possibilitam a realização do processo de Extração da Informação, apresentando um desempenho adequado para a sua aplicação em situações práticas do mundo real.

Quadro 12: Questão de pesquisa.

O uso integrado dos mecanismos de inferência da Ontology Web Language (OWL) em conjunto com a ontologia de domínio e as regras linguísticas permitem a composição de um artefato computacional que possibilite a Extração da Informação com desempenho adequado?

Para chegar-se à resposta da questão de pesquisa, buscou-se um embasamento teórico consistente, profundo e abrangente, o qual é apresentado de forma resumida no capítulo 2. A fundamentação teórica apresentada é resultante do levantamento do estado da arte no que tange a EI, realizado através da leitura de trabalhos acadêmicos. A partir desta leitura, foram selecionados e apresentados no capítulo 3 desta dissertação os trabalhos que mais influenciaram a solução aqui proposta.

No capítulo 4 apresenta-se o modelo para a Extração de Informações a partir de documentos contendo textos em linguagem natural. Complementando o objetivo principal, elaborou-se um modelo que propõe uma abordagem em duas etapas, composta por uma Fase Linguística, cujo objetivo é a criação de uma ontologia de domínio baseada em estudo de corpora e a elaboração de regras da ontologia de extração, seguida de uma Fase Computacional, onde o processo de EI é efetivamente realizado, objetivando-se a identificação de conceitos representados na ontologia de domínio.

Sendo a implementação do processo automatizado de extração de conceitos a partir de corpus de domínio o foco principal deste trabalho, apresenta-se detalhadamente a metodologia da Fase Computacional, a qual tem como entrada um corpus anotado e como saída uma Base de Conhecimentos contendo a localização dos conceitos referenciados no corpus analisado.

A avaliação do desempenho da proposição deste trabalho, apresentada no capítulo 5, foi realizada pela experimentação da metodologia sobre um corpus do domínio jurídico, o qual compõe-se de um conjunto de textos com quase 620 mil palavras. A aferição do desempenho do processo de extração foi realizada manualmente por especialistas, considerando-se como quesitos de desempenho as métricas de *Acurácia, Precisão* e *Revocação* alcançadas no reconhecimento dos eventos jurídicos contidos na ontologia de domínio ODomJurBR.

As atividades desenvolvidas no âmbito desta dissertação, realizadas com vistas a obter-se a resposta para a questão de pesquisa, resultaram em contribuições científicas efetivas e em possibilidades de continuidade da pesquisa aqui iniciada, ambas apresentadas nas próximas seções deste capítulo.

6.1 Contribuições Científicas

Com relação ao legado científico, entende-se que como consequência das pesquisas realizadas para responder à questão de pesquisa foram produzidas algumas contribuições científicas e tecnológicas.

Primeiramente, a disponibilização de um modelo para implementação de um processo de EI que propõe, no seu fluxo operacional, um melhor aproveitamento dos esforços realizados durante a fase de análise de corpus.

Via de regra, a fase de análise de corpus visa somente a criação da ontologia de domínio, mas no modelo aqui sugerido são produzidas também anotações que visam a otimização do desempenho das regras de extração.

A anotação do corpus visando a geração de regras permite ao especialista em representação do conhecimento focar seus esforços na elaboração de RACs que obtenham alto desempenho de extração e não na análise linguística de frases, pois não é esta a sua especialidade.

Outra contribuição deste trabalho está relacionada à independência da terminologia adotada para a representação dos conceitos na ontologia de domínio viabilizada pela abordagem de EI aqui desenvolvida.

Diferentemente de outras implementações de EI baseadas em ontologias, que normalmente utilizam a ontologia como uma lista de termos do domínio, a abordagem proposta neste trabalho reforça a autonomia e independência do processo de estruturação da ontologia de domínio em relação ao processo de extração.

A principal vantagem da independência da terminologia adotada na ontologia de domínio centra-se no fato de que a preocupação ao projetar-se a sua estrutura deve ser a representação dos conceitos do domínio e não as diferentes formas lexicais dos conceitos do domínio.

Na proposição deste trabalho, as diversas formas lexicais dos conceitos são expressas pelas RACs formalizadas na ontologia de extração. Além de não interferir na composição terminológica da ontologia de domínio, pode-se assim associar aos conceitos representações terminológicas totalmente díspares, como por exemplo a associação entre a frase "*Subiram os autos*" e o conceito de *Movimentação_Processual* do domínio jurídico.

Também considera-se uma contribuição deste trabalho a proposição de um processo de EI com lógica unificada sob a forma de ontologia, pois a representação dos conceitos e das regras de extração, respectivamente formalizadas pelo especialista em representação de conhecimento nas ontologias de domínio e de extração, reduz a quantidade de conhecimentos e especialistas necessários para a aplicação do modelo proposto.

Outro reflexo da lógica unificada para o processo de EI é o encapsulamento do processo de EI sob a forma de ontologias, herdando daí todas as vantagens inerentes ao uso desta representação, tais como o compartilhamento, reutilização, representação de uma mesma conceitualização em várias línguas e extensão.

O uso do paradigma declarativo de programação baseado em lógica para a formalização das regras de extração com níveis de flexibilidade, expressividade e

generalização adequados para a implementação dos processos de EI e a apresentação de resultados é uma contribuição significativa deste trabalho.

Embora encontrem-se trabalhos que utilizem o sistema inferencial do OWL para a implementação do processo de EI (DEDEK; VOJTAS, 2011), não são relatados nem o desempenho obtido do processo de extração nem o número de conceitos a serem identificados.

Por fim, a utilização implícita das relações representadas na ontologia de domínio no transcorrer do processo de EI demonstraram um alto nível de integração entre a conhecimento representado na ontologia de domínio e o processo de extração representado nas regras da ontologia de extração.

A possibilidade de utilização implícita das relações declaradas na ontologia de domínio se dá pela união desta com a ontologia de extração, unificando em uma única ontologia todos os conceitos, relações e restrições do domínio e das regras do processo de extração.

A declaração de equivalência entre as classes *Réu* e *Acusado* e o efeito sobre a RAC para identificação do evento de *Absolvição* é um exemplo de utilização implícita das relações de sinonímia da ontologia de domínio no processo de extração, possibilitando o reconhecimento da absolvição tanto do réu quanto do acusado através de somente uma regra.

Além das contribuições acima elencadas, cabe ainda citar a elaboração de um artigo relativo a este trabalho, o qual foi aprovado para apresentação no evento "First Knowledge Discovery in Ontologies", um workshop para relato de experiências sobre criação de ontologias, a ser realizado conjuntamente ao evento WI-IAT 2013 (The 2013 IEEE/WIC/ACM International Conference on Web Intelligence and International Conference on Intelligent Agent Technology).

6.2 Trabalhos Futuros

Realizados os experimentos, confirmada a viabilidade da proposta e verificado o desempenho pretendido, despontam possibilidades de continuidade deste trabalho, seja do ponto de vista do aprofundamento do potencial da abordagem através de novos experimentos, seja pela investigação de novos campos de pesquisa para a sua aplicação.

Verificou-se, principalmente no transcorrer do segundo experimento, a necessidade da implementação de um ambiente integrado para o desenvolvimento das regras de extração, o qual instrumentalizasse o especialista em representação do conhecimento na formalização das RACs.

A disponibilização de um sistema que permitisse verificar em tempo real a efetividade e abrangência de uma RAC em relação aos documentos do corpus contribuiria sensivelmente para a qualidade e velocidade de formalização das regras de extração, impactando diretamente sobre o desempenho do processo de EI.

A realização de um experimento que envolvesse os demais especialistas do grupo de pesquisa, talvez em uma nova área de domínio, seria um próximo passo em direção à adoção do modelo aqui sugerido. Tal possibilidade demandaria este ambiente integrado para o

desenvolvimento das RACs, sendo assim uma ferramenta essencial para o aprofundamento das pesquisas do potencial da abordagem aqui sugerida.

Um procedimento operacional, vislumbrada mas não implementada neste trabalho, seria a conversão dos documentos linguisticamente anotados no formato *Árvore Deitada* diretamente para OWL. Além do efeito colateral de reduzir o tempo necessário para a conversão dos documentos do corpus para OWL, teria como principal benefício a preservação de informações linguísticas produzidas pelo parser Palavras e que sabidamente são perdidas na conversão para o formato TIGER-XML.

Em sentenças mais complexas, a identificação dos conceitos da ontologia de domínio demanda regras linguísticas mais elaboradas, regras estas que por sua vez dependem da identificação precisa das relações que os componentes do sintagma tem entre si. Tais relações são perdidas no momento de converter-se as anotações linguísticas para TIGER-XML. A perda destas informações linguísticas tem como impacto a redução da precisão das RACs e, por consequência, a diminuição do desempenho do processo de EI.

Experimentos para verificar-se mais profundamente os efeitos da reutilização implícita das relações e restrições definidas na ontologia de domínio sobre as regras de extração é outra área de pesquisa que instiga a curiosidade, pois no estudo de caso aqui realizado se verificaram resultados positivos, no entanto, não se sabe qual o resultado em ontologias de domínio mais extensas e complexas.

Uma outra linha de pesquisa a ser investigada em relação a abordagem sugerida neste trabalho é o uso de diversas camadas de anotação para a elaboração das RACs. O uso do modelo de dados POWLA possibilita que sejam representadas no arquivo OWL várias camadas de anotação, o que viabilizaria o desenvolvimento de regras de extração com base em informação abstrata de múltiplas origens, tais como *parsers* de correferência, sistema de reconhecimento automático de entidades nomeadas, etc.

Tem-se como perspectiva em relação a elaboração de regras baseadas em múltiplas camadas de anotação a ampliação o poder de representação das RACs, resultando em aumento no desempenho da *Acurácia*, da *Precisão* e da *Revocação* do processo de EI.

Outra possibilidade a ser explorada é a conversão das regras de extração em DL ou SWRL para SPARQL. A possibilidade de armazenamento e aplicação das regras de extração sob a forma de consultas SPARQL teria como principal vantagem um melhor desempenho na aplicação da metodologia aqui desenvolvida na extração de informação a partir de documentos ou repositórios realmente muito extensos.

Além do uso de múltiplas camadas de anotação, outra área possível de ser explorada utilizando-se a abordagem proposta deste trabalho seria a de utilização do modelo conceitual OLiA para a elaboração das regras de extração.

A adoção do OLiA para a elaboração das regras teria como vantagem tornar as regras genéricas e independentes do parser utilizado, o modelo conceitual atuaria como uma camada de abstração para o conjunto de etiquetas de anotação (*tagset*) do parser.

Esta camada de abstração, em princípio, possibilitaria que as regras de extração fossem independentes do *parser*, pois utilizariam o *tagset* abstrato do modelo conceitual. Na prática, as regras de extração poderiam utilizar anotações geradas por qualquer parser cujo padrão de anotação estivesse representado em uma ontologia de Anotação do OLiA.

O primeiro passo no sentido de verificar-se a possibilidade do uso combinado das RACs com o modelo conceitual OLiA seria o desenvolvimento das ontologias de Anotação e de Ligação para a representação do tagset utilizado pelo parser Palavras, o Portuguese VISL symbol set, sendo este outro interessante experimento a desenvolver-se.

Uma outra linha de pesquisa, a qual extrapola as fronteiras da fundamentação teórica apresentada neste trabalho, estaria relacionada com a investigação de possíveis intersecções conceituais entre a desambiguação de significados baseada em regras de extração formalizadas em OWL (as RACs) e a definição de papéis baseada na teoria de significados chamada *Frames Semânticos*.

Seja pelo estudo da possibilidade de utilização das bases de dados do projeto FrameNet para a geração automatizada de RACs, seja pela investigação da viabilidade de formalização dos frames semânticos através das RACs, ambas frentes de pesquisa são muito instigantes e despertam a curiosidade científica.

Percebe-se, mesmo que ainda muito intuitivamente, uma sobreposição de conceitos e objetivos entre as RACs e os Frames Semânticos. São especulações e conjecturas, carentes ainda de uma investigação científica mais profunda, mas que vislumbram horizontes promissores e desafiantes.

7 REFERÊNCIA BIBLIOGRÁFICA

ALMEIDA Fº, José Carlos de Araújo. *Processo eletrônico e teoria geral do processo eletrônico: a informatização judicial no Brasil*. São Paulo: Forense, 2010.

ALVES, Isa Mara da Rosa. *O uso da Semântica Verbal em Sistemas de Extração da Informação: a Construção de uma Ontologia de Domínio Jurídico.* 2005. Dissertação (Mestrado em Linguística Aplicada) – Programa de Pós-Graduação em Linguística Aplicada. UNISINOS, São Leopoldo, RS, 2005.

AMARDEILH, Florence; LAUBLET, Philippe; MINEL, Jean Luc. *Document Annotation and Ontology Population from Linguistic Extractions***.** Proceedings of Knowledge Capture (KCAP), Banff, 2005.

ANDROUTSOPOULOS, Ion; MALAKASIOTIS, Prodromos. A survey of paraphrasing and textual entailment methods. arXiv preprint arXiv:0912.3747, 2009.

ANTONIOU, G.; HARMELEN, F. Web Ontology Language: OWL. In: *Handbook on Ontologies SE*, Staab, S., Studer, R. (eds.). Second Edition. International Handbooks on Information System. Springer-Verlag, Berlin, Germany. 2009.

BAADER, Franz; HORROCKS, Ian; SATTLER, Ulrike. Description Logic. In: *Handbook on Ontologies SE*, STAAB, S., STUDER, R. (eds.). Second Edition. International Handbooks on Information System. Springer-Verlag, Berlin, Germany. 2009.

BAADER, Franz; SATTLER, Ulrike. *An overview of tableau algorithms for description logic.* Studia Logica, 69(1):5–40, 2001.

BERNERS-LEE, Tim; FIELDING, Roy; MASINTER, Larry. *Uniform resource identifiers (URI): generic syntax*. 1998. Disponível em: <http://www.hjp.at/doc/rfc/rfc3986.html>. Acesso em: 30 Oct. 2013.

BERNERS-LEE, Tim; HENDLER, James; LASSILA, Ora. *The Semantic Web*. Scientific American (May 2001), n. 284, p. 35–40, 2001.

BICK, Eckhardt. *The Parsing System Palavras: Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework.* Aarhus University Press, 2000.

BOER, Alexander; DI BELLO, Marcello; VAN DEN BERG, Kasper; GORGON, Tom; FÖRHÉCZ, András; VAS, Réka; KLARMAN, Szymon; HOEKSTRA, Rinke. *Specification of the Legal Knowledge Interchange Format*: Deliverable 1.1: summary. Estrella Project. 2006. Disponível em: <http://dare.uva.nl/document/167498>. Acesso em: 30 Oct. 2013

BOER, Alexander; WINKELS, Radboud; VITALI, Fabio. *MetaLex XML and the Legal Knowledge Interchange Format*. Computable Models of the Law. Lecture Notes in Artificial Intelligence 4884, pp. 21-41, Springer Verlag, Berlin. 2008.

BORST, W. N. *Construction of engineering ontologies*. Tese (Doutorado). 1997. Disponível em: <http://www.ub.utwente.nl/webdocs/ inf/1/t0000004.pdf>. Acesso em: 25 Jul. 2013

BRANSFORD-KOONS, Geoffrey R. *Dynamic semantic annotation of California case law*. PhD dissertation, San Diego State University, 2005.

BRAY, Tim; PAOLI, Jean; SPERBERG-MCQUEEN, C. M.; MALER, Eve; YERGEAU, François; COWAN, John. *Extensible markup language (XML) 1.1 (Second Edition)*, W3C recommendation 16 August 2006. Technical Report REC-xml11-20060816, World Wide Web Consortium, 2006.

BREUKER, Joost; HOEKSTRA, Rinke. *DIRECT: Ontology-based Discovery of Responsibility and Causality in Legal Case Descriptions*. In: Proceedings of The 17th JURIX (Berlin, DE, 2004), p. 59– 68. Amsterdam *et al.*: IOS Press, 2004.

BRICKLEY, D.; GUHA, R. V. *RDF Vocabulary Description Language 1.0: RDF Schema.* W3C Recommendation, 2004. Disponível em: <http://www.w3.org/TR/rdf-schema>. Acesso em: 25 Jul. 2013.

BRUNINGHAUS, S.; ASHLEY, K. *Improving the Representation of Legal Case Texts with Information Extraction Methods.* Proceedings of the 8th International Conference on Artificial Intelligence and Law (ICAIL '01), pages 42-51, ACM Press, 2001.

BUITELAAR, Paul; CIMIANO, Philipp; HAASE, Peter; SINTEK, Michael. *Towards linguistically grounded ontologies*. In: The Semantic Web: Research and Applications. Springer Berlin Heidelberg, 2009. p. 111-125

BUYKO, Ekaterina; CHIARCOS, Christian; PAREJA-LORA, Antonio. *Ontology-based interface specifications for a NLP pipeline architecture*. In: Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008). 2008.

CALVANESE, Diego; GIACOMO, Giuseppe de; LENZERINI, Maurizio; NARDI, Daniele. *Reasoning in expressive description logic*. In: ROBINSON, Alan; VORONKOV, Andrei (eds.). Handbook of Automated Reasoning, chapter 23. Elsevier, Amsterdam. 2001. p. 1581–1634

CASTILLO, Jaime A. Reinoso; SILVESCU, A.; CARAGEA, D.; PATHAK, J.; HONAVAR, V. G. *Information extraction and integration from heterogeneous, distributed, autonomous information sources — a federated ontology-driven query-centric approach. In Information Reuse and Integration*, 2003. IRI 2003. IEEE International Conference on (pp. 183-191). IEEE. 2003.

CHAKRABARTI, Soumen. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan-Kauffman Publishers, San Francisco, CA, USA, 2002.

CHIARCOS, Christian. *An ontology of linguistic annotations*. LDV Forum, 23(1), p. 1–16, 2008.

CHIARCOS, Christian. *Grounding an ontology of linguistic annotations in the Data Category Registry*. In: LREC Workshop on Language Resource and Language Technology Standards (LT&LTS), Valetta, Malta, May, 2010. p. 37–40.

CHIARCOS, Christian; ERJAVEC, Tomaz. *OWL/DL formalization of the MULTEXT - East morphosyntactic specifications*. ACL HLT 2011: 11, 2011.

CHIARCOS, Christian. *POWLA: Modeling linguistic corpus in OWL/DL*. Proceedings of 9th Extended Semantic Web Conference (ESWC2012), 2012a.

CHIARCOS, Christian. *Ontologies of linguistic annotation: Survey and perspectives*. In: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC), p. 303-310. 2012b.

CHIARCOS, Christian. *A generic formalism to represent linguistic corpora in RDF and OWL/DL.* In: 8th International Conference on Language Resources and Evaluation (LREC 2012). 2012c.

CLARK, James (ed.). *XSL Transformations (XSLT)*. W3C Recommendation (16 November 1999). 1999. Disponível em: <http://www.w3.org/TR/xslt>. Acesso em: 22 jul. 2013.

COWIE, Jim; LEHNERT, Wendy. *Information Extraction*. Communications of the ACM, v. 39, n. 1, Jan. 1996. p. 80 – 91. ACM New York, NY, USA. 1996.

CUNNINGHAM, Hamish; MAYNARD, Diana; BONTCHEVA, Kalina; TABLAN, Valentin. *GATE: A framework and graphical development environment for robust NLP tools and applications*. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02), 2002.

DE MARNEFFE, Marie-Catherine; MACCARTNEY, Bill; MANNING, Christopher D. Generating typed dependency parses from phrase structure parses. In: Proceedings of LREC, vol. 6, p. 449-454. 2006.

DEDEK, Jan; VOJTAS, Peter. *Semantic Annotation Semantically: Using a Shareable Extraction Ontology and a Reasoner*. In: Proceedings of SEMAPRO 2011, The Fifth International Conference on Advances in Semantic Processing, p. 29-34, Lisbon, 2011.

EL-GOHARY, Nora M.; EL-DIRABY, Tamer E. *Domain ontology for processes in infrastructure and construction*. Journal of Construction Engineering and Management, v. 136, n. 7, p. 730-744. 2010.

EMBLEY, David W. *Toward semantic understanding: an approach based on information extraction ontologies*. In: Proceedings of the 15th Australasian database conference-Volume 27, pp. 3-12. Australian Computer Society, Inc., 2004.

EMBLEY, David W.; TAO, Cui; LIDDLE, Stephen W. *Automating the extraction of data from HTML tables with unknown structure*. Data & Knowledge Engineering 54, no. 1: p. 3-28. 2005.

FARRAR, Scott; LANGENDOEN, D. Terence. *An OWL-DL Implementation of Gold*. In: Linguistic Modeling of Information and Markup Languages. Springer Netherlands, 2010. p. 45-66.

FRIEDL, Jeffrey E. F. Mastering Regular Expressions (Third Edition). O´Reilly, Sebastopol, CA, USA. 2006.

GRUBBER, Thomas R. *ONTOLINGUA: A Mechanism to Support Portable Ontologies*. Knowledge System Laboratory. CA: Stanford University, 1992.

GRUBBER, Thomas R. *A translation approach to portable ontologies*. Knowledge Acquisition, v. 5, n. 2, p. 199-220, 1993.

GRUBBER, Thomas R. *Toward principles for the design of ontologies used for knowledge sharing*. In: International Journal of Human-Computer Studies, v. 43, n. 4-5, p. 907-928. 1995.

GUARINO, Nicola. *Semantic matching: Formal ontological distinctions for information organization, extraction, and integration*. In: Information Extraction A Multidisciplinary Approach to an Emerging Information Technology. Springer Berlin Heidelberg, 1997. 139-170.

GUARINO Nicola; OBERLE, Daniel; STAAB, Steffen. *What Is an Ontology?* In: Handbook on Ontologies SE, Staab, S., Studer, R. (eds.). Second Edition. International Handbooks on Information System. Springer-Verlag, Berlin, Germany. 2009. p. 1-17.

GUIMARÃES, José Augusto Chaves; BASÍLIO, Marisa Bräscher; DE SORDI, Neide Alves Dias. *Manual de Indexação de Jurisprudência da Justiça Federal.* Conselho da Justiça Federal, Brasília. 1996.

HALVORSON, Marty. OASIS Legal XML Member Section, Electronic Court Filing, Technical Committee, Electronic Court Filing 1.1, Proposed Standard. 22 July 2002. http://www.oasis-open.org/committees/legalxml-courtfiling/documents/22072002cf1-1.pdf. Acesso em: 25 Jul. 2013.

HARDOUIN, Ronan. Le sens des responsabilitiés en matiere de contrats informatiques. Vol. 1. Technical report, Livrable LISE. 2009.

HAYES, Patrick (ed.). *RDF Semantics*. W3C Recommendation (10 February 2004). 2004. Disponível em: <http://www.w3.org/TR/rdf-mt/>. Acesso em: 25 Jul. 2013.

HITZLER, Pascal; PARSIA, Bijan. *Ontologies and Rules*. In: Handbook on Ontologies SE, Staab, S., Studer, R. (eds.). Second Edition. International Handbooks on Information System. Springer-Verlag, Berlin, Germany. 2009.

HORRIDGE, M.; KNUBLAUCH, H.; RECTOR, A.; STEVENS, R.; WROE, C. *Protégé OWL Tutorial*, The University Of Manchester, Stanford University. 2004. Disponível em: <http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/>. Acesso em: 25 Jul. 2013.

HORROCKS, Ian. *DAML+OIL: a description logic for the Semantic Web*. IEEE Data Eng Bull, v. 25, n. 1, p. 4–9, 2002.

HORROCKS, Ian; KUTZ, O.; SATTLER, Ulrike. *The even more irresistible SROIQ*. In: Proceedings KR, 2006.

HORROCKS, Ian; PATEL-SCHNEIDER, Peter F. *A proposal for an OWL rules language*. In: Proceedings of the Thirteenth International World Wide Web Conference (WWW 2004). ACM, New York, 2004, pages 723–731.

HORROCKS, Ian; PATEL-SCHNEIDER, Peter F.; BOLEY, Harold; TABET, Said; GROSOF, Benjamin; DEAN, Mike. *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. W3C Recommendation (21 Maio 2004). 2004. Disponível em: <http://www.w3.org/Submission/SWRL/>. Acesso em: 25 Jul. 2013.

HORROCKS, Ian; SATTLER, Ulrike; TOBIES, Stephan. *Reasoning with individuals for the description logic SHIQ*. In David McAllester, editor, Proceedings of the 17th Int. Conference on Automated Deduction (CADE 2000), volume 1831 of Lecture Notes in Computer Science, pages 482–496. Springer, Berlin, 2000.

JIJKOUN, Valenti; RIJKE, Maarten de; MUR, Jori. *Information extraction for question answering: Improving recall through syntactic patterns*. In: Proceedings of the 20th international conference on Computational Linguistics, Association for Computational Linguistics, p. 1284, 2004.

KARA, Soner; ALAN, Özgür; SABUNCU, Orkunt; AKPINAR, Samet; CICEKLI, Nihan K.; ALPASLAN, Ferda N. *An ontology-based retrieval system using semantic indexing*. In: Information Systems 37, no. 4: 294-305, 2012.

KIPPER, Karin; KORHONEN, Anna; RYANT, Neville; PALMER, Martha. A large-scale classification of English verbs. Language Resources and Evaluation, 42(1): p. 21–40, 2008.

KOLAITIS, Phokion G.; VARDI, Moshe Y. *Conjunctive-Query Containment and Constraint Satisfaction.* In: Journal of Computer and System Sciences 61: 302–332. 2000.

KÖNIG, E.; Lezius, W. *The TIGER Language – A Description Language For Syntax Graphs.* Formal definition, Technical Report, 2003.

KUBA, Martin. *OWL 2 and SWRL Tutorial*. Institute of Computer Science, Masaryk University, Brno. Disponível em: <http://dior.ics.muni.cz/~makub/owl/>. Acesso em: 25 Jul. 2013.

LACLAVIK, Michal; ŠELENG, Martin; CIGLAN, Marek; HLUCHÝ, Ladislav. "Ontea: Platform for pattern based automated semantic annotation." Computing and Informatics 28, no. 4 (2012): 555-579. 2012.

LABSKÝ, Martin; SVÁTEK, Vojtech; NEKVASIL, Marek. *Information Extraction Based on Extraction Ontologies: Design, Deployment and Evaluation*. ONTOLOGY-BASED INFORMATION EXTRACTION SYSTEMS (OBIES 2008) (2008): 9. 2008.

LABSKÝ, Martin; SVÁTEK, Vojtěch; NEKVASIL, Marek; RAK, Dušan. *The Ex Project: Web Information Extraction using Extraction Ontologies*. In: Knowledge Discovery Enhanced with Semantic and Social Information, p. 71-88. Springer Berlin Heidelberg, 2007.

LENCI, A.;MONTEMAGNI, S.; PIRRELLI, V.; VENTURI, G. *NLP-Based Ontology Learning From Legal Texts – A Case Study.* Proceedings of LOAIT, 2007.

LESMO, Leonardo. *The rule-based parser of the NLP group of the University of Torino*. Intelligenza artificiale 2, no. 4, p. 46-47. 2007.

LESMO, Leonardo; MAZZEI, Alessandro; PALMIRANI, Monica; RADICIONI, Daniele P. TULSI: an NLP system for extracting legal modificatory provisions. Artificial Intelligence and Law. v. 21, p. 139-172. Springer Netherlands. 2013.

LIDDY, Elizabeth D. *Natural Language Processing.* In: Encyclopedia of Library and Information Science, 2nd ed. New York: Marcel Decker, Inc., 2003.

LIMA, João Alberto de Oliveira; CICILIATI, Fernando (eds.). *LexML Brasil. 2008* – Disponível em: <http://projeto.lexml.gov.br/documentacao/Apresentacao.pdf>. Acesso em: 25 jul. 2013.

LUPO, Caterina; VITALI, Fabio; FRANCESCONI, Enrico; PALMIRANI, Monica; WINKELS, Radboud; DE MAAT, Emile; BOER, Alexander; MASCELLANI, Paolo. *Estrella D3.1*. Faculty of Law, University of Amsterdam, Amsterdam. 2007. Disponível em: <http://www.estrellaproject.org/doc/D3.1-General-XML-formats-For-Legal-Sources.pdf>. Acesso em: 25 Jul. 2013

MAZZEI, Alessandro; RADICIONI, Daniele P.; BRIGHI, Raffaella. *NLP-Based Extraction of Modificatory Provisions Semantics.* Proceedings of the 12th International Conference on Artificial Intelligence and Law (ICAIL), p. 50–57, Barcelona, Spain, 2009.

MAAREK, Manuel. On the extraction of decisions and contributions from summaries of Legal IT Contract Cases. In: LREC 2010 Workshop on Semantic Processing of Legal Texts proceedings (SPLeT). 2010.

MINGHELLI, Thaís Domingues. *A Relação de Meronímia em uma Ontologia Jurídica.* 2011. Dissertação (Mestrado em Linguística Aplicada) – Programa de Pós-Graduação em Linguística Aplicada. UNISINOS, São Leopoldo, RS, 2011.

MOENS, Marie-Francine; UYTTENDAELE, Caroline; DUMORTIER, Jos. *Information Extraction from Legal Texts: The Potential of Discourse Analysis.* In: International Journal of Human-Computer Studies, v. 51, p. 1155–1171, 1999.

MOENS, Marie-Francine. Improving access to legal information: How drafting systems help. In: Information Technology and Lawyers. Springer Netherlands, p. 119-136, 2006.

MOMMERS, L. *A Knowledge-Based Ontology of the Legal Domain*. In: Second International Workshop on Legal Ontologies, JURIX 2001, Amsterdam, Netherlands, December. 2001.

MOTIK, Boris. *Reasoning in Description Logic using Resolution and Deductive Databases*. PhD thesis, Universitat Karlsruhe (TH), Germany, 2006.

MOTIK, Boris; GRAU, Bernardo Cuenca; HORROCKS, Ian; WU, Zhe; FOKOUE, Achille; LUTZ, Carsten (eds.). *OWL 2 Web Ontology Language: Profiles*. W3C Recommendation, 2009. Disponível em: <http://www.w3.org/TR/owl2-profiles/>. Acesso em: 25 Jul. 2013.

MOTIK, Boris; PATEL-SCHNEIDER, Peter F.; GRAU, Bernardo Cuenca (eds.). *OWL 2 Web Ontology Language: Direct Semantics.* W3C Recommendation, 2009. Disponível em: <http://www.w3.org/TR/owl2-direct-semantics/>. Acesso em: 25 Jul. 2013.

MOTIK, Boris; SATTLER, Ulrike; STUDER; Rudi. *Query answering for OWL-DL with rules*. Journal of Web Semantics, v. 3, n. 1, p. 41–60. 2005.

NARDI, Daniele; BRACHMAN, Ronald J. *An introduction to Description Logic*. In: BAADER, Franz; McGuinness, Deborah L.; NARDI, Danieli; PATEL-SCHNEIDER, Peter F. (eds.). The Description Logic Handbook: Theory, Implementation, and Applications. [S. l.]: Cambridge University Press, 2003. p. 5-44.

NÉDELLEC, Claire; NAZARENKO, Adeline; BOSSY; Robert. *Information Extraction*. In: Handbook on Ontologies SE, Staab, S., Studer, R. (eds.). Second Edition. International Handbooks on Information System. Springer-Verlag, Berlin, Germany. 2009.

NOY, Natasha; RECTOR, Alan. *Defining N-ary Relations on the Semantic Web*. World Wide Web Consortium (2006-04-12). Disponível em: <http://www.w3.org/TR/swbp-n-aryRelations>. Acesso em: 25 Jul. 2013.

NUTTER, J. Terry. *Epistemology*. In: S. Shapiro, editor, Encyclopedia of Artificial Intelligence. John Wiley & Sons, New York, 1987.

OWL-WG W3C. *OWL 2 Web Ontology Language Document Overview*, 2009-10-27. Disponível em: <http://www.w3.org/TR/owl2-overview/>. Acesso em: 30 Oct. 2013.

PALMIRANI, Monica; BRIGHI, Raffaella; CASANOVAS, Pompeu; PAGALLO, Ugo; SARTOR, Giovani; AJANI, Gianmaria. *Model Regularity of Legal Language in Active Modifications*. In: AI Approaches to the Complexity of Legal Systems. Complex Systems, the Semantic Web, Ontologies, Argumentation, and Dialogue, Book Series Title: Lecture Notes in Computer Science, v. 6237, p. 54-73. Springer Berlin / Heidelberg, 2010.

PARK, Jack; HUNTING, Sam. *XTM Topic Maps: Creating and using Topic Maps for the Web*. Boston: Addison Wesley, 2003.

REALE, Miguel. *Lições preliminares de direito*. Saraiva, 1977.

RILOFF, Ellen. *Automatically Generating Extraction Patterns from Untagged Text*. In: Proceedings of the Thirteenth National Conference on Artificial Intelligence, p. 1044-1049. 1996.

RILOFF, Ellen, *Information Extraction as a stepping stone toward story understanding. Understanding Language Understanding: computational models of reading*, MIT Press. p. 435-460. 1999.

RILOFF, Ellen; PHILLIPS, William. *An Introduction to the Sundance and AutoSlog Systems.* Technical Report UUCS-04-015, School of Computing, University of Utah, 2004.

RILOFF, Ellen; SCHELER, Gabriele (eds). Connectionist, statistical and symbolic approaches to learning for natural language processing. Vol. 1040. Springer Verlag, 1996.

SANG, Erik Tjong Kim; HOFMANN, Katja. *Lexical patterns or dependency patterns: which is better for hyperonym extraction?*. In: Proceedings of the Thirteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, pp. 174-182, 2009.

SANTOS, E. F. *Manual de direito processual civil*. v. 1. n. 9. Ed. São Paulo: Saraiva, 2002.

SARAVANAN, M.; RAVINDRAN, B.; RAMAN, S. *Improving Legal Information Retrieval Using an Ontological Framework.* Artificial Intelligence and Law. Springer Netherlands, v. 17, n. 2, p. 101-124, 2009.

SCHNEIDER, Michael (ed.). *OWL 2 Web Ontology Language: RDF Based Semantics*. W3C Recommendation (27 October 2009), Disponível em: <http://www.w3.org/TR/owl2-rdf-based-semantics>. Acesso em: 25 Jul. 2013.

SILVA, De Plácido e. SLAIBI Fº, Nagib; CARVALHO, Gláucia (atualizadores). *Vocabulário Jurídico*, 28ª ed., RJ: Editora Forense, 2010.

SITTHISARN; Siraya; LAU, Lydia; DEW, Peter M. *Semantic keyword search for expert witness discovery*. International Conference on Semantic Technology and Information Retrieval, Putrajaya, Malaysia, 2011.

SOWA, John F. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Pacific Grove, CA: Brooks/Cole, 2000.

WIMALASURIYA, Daya C.; DOU, Dejing, *Using Multiple Ontologies in Information Extraction*. CIKM, 09, ACM Press: Hong Kong, p. 235-245, 2009.

WIMALASURIYA, Daya C.; DOU, Dejing, *Ontology-based information extraction: an introduction and a survey of current approaches.* Journal of Information Science, vol. 36, no. 3, pp. 306–323, 2010.

WYNER, Adam. *Towards annotating and extracting textual legal case elements*. Informatica e Diritto 19, no. 1-2 (2010): p. 9-18. 2010.

WYNER, Adam; PETERS, Wim. *On rule extraction from regulations*. In: Proceedings of the 24th International Conference on Legal Knowledge and Information Systems (JURIX 2011), University of Vienna, 2011.

ZHANG, Jiansong; EL-GOHARY, Nora. *Extraction of Construction Regulatory Requirements from Textual Documents Using Natural Language Processing Techniques*. In: *Proceedings Computational Civil Engineering*, p. 453-460. 2012.

APÊNDICE A EXEMPLO DE ACÓRDÃO

ESTADO DO RIO GRANDE DO SUL
PODER JUDICIÁRIO
TRIBUNAL DE JUSTIÇA

NOP
Nº
2012/CRIME

EMBARGOS INFRINGENTES. CRIMES SEXUAIS CONTRA VULNERÁVEIS. ESTUPRO DE VULNERÁVEL. TENTATIVA RECONHECIDA.
As provas colhidas no curso da instrução revelam que a ação delitiva foi frustrada pela intervenção de uma testemunha no momento em que o embargante realizava atos de esfregação de seu pênis nas nádegas do ofendido – criança com nove anos de idade ao tempo do abuso.
Assim, em virtude de circunstância alheia à vontade do agente, este foi impedido de realizar todos os atos libidinosos que pretendia, o que determina o reconhecimento da forma tentada da infração, nos moldes propostos no voto minoritário quando do julgamento do recurso de apelação.
EMBARGOS INFRINGENTES ACOLHIDOS.

EMBARGOS INFRINGENTES E DE NULIDADE — QUARTO GRUPO CRIMINAL

Nº — COMARCA DE

EMBARGANTE

EMBARGADO

ACÓRDÃO

Vistos, relatados e discutidos os autos.

Acordam os Desembargadores integrantes do Quarto Grupo Criminal do Tribunal de Justiça do Estado, face ao empate, em acolher os embargos infringentes, decisão esta, tomada com base no artigo 21, § 2º, I do RITJERGS.

Custas na forma da lei.

Participaram do julgamento, além da signatária, os eminentes Senhores **DES. DANÚBIO EDON FRANCO (PRESIDENTE), DES. CARLOS ALBERTO ETCHEVERRY, DES.ª FABIANNE BRETON BAISCH,**

ESTADO DO RIO GRANDE DO SUL
PODER JUDICIÁRIO
TRIBUNAL DE JUSTIÇA

NOP
Nº
/CRIME

DES.ª ISABEL DE BORBA LUCAS E DES.ª LAURA LOUZADA JACCOTTET.

Porto Alegre,

DES.ª NAELE OCHOA PIAZZETA,
Relatora.

RELATÓRIO

DES.ª NAELE OCHOA PIAZZETA (RELATORA)

Trata-se de embargos infringentes opostos por , em razão de decisão não unânime exarada pela Oitava Câmara Criminal, pela qual desprovido o apelo defensivo, vencido o Desembargador Dálvio Leite Dias Teixeira, que o provia em parte, ao efeito de reconhecer a forma tentada da infração e reduzir a pena.

Com base no voto minoritário, a Defensoria Pública pugna pela reforma do comando majoritário para prevalecer a tese vencida.

Recebidos os embargos, manifesta-se o Ilustre Procurador de Justiça pelo seu desacolhimento.

Conclusos para julgamento.

É o relatório.

VOTOS

DES.ª NAELE OCHOA PIAZZETA (RELATORA)

Eminentes Colegas.

Como relatado, trata-se de embargos infringentes opostos contra decisão não unânime exarada pela Oitava Câmara Criminal, no julgamento da Apelação Crime nº , pela qual desprovido o

APÊNDICE B REGRAS DE EXTRAÇÃO UTILIZADAS NOS EXPERIMENTOS

```
# ABS01 - ABSOLVIÇÃO ABS-01
#   Frase: absolver o acusado
Absolver(?tVerbo),
hasParent(?tVerbo, ?cl),
isSourceOf(?cl, ?relOd),
has_label(?relOd,"Od"^^string),
hasTarget(?relOd, ?npAcusado),
hasDescendent(?npAcusado,?tAcusado),
Acusado(?tAcusado) -> Absolvição(?cl)

# ABS02 - ABSOLVIÇÃO (voz passiva)
#   Frase: o réu ... foi absolvido
Absolver(?verbo),
hasParent(?verbo, ?vpPass),
isSourceOf(?vpPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
hasParent(?vpPass, ?fcl),
isSourceOf(?fcl, ?relSujPass),
has_label(?relSujPass, "S"^^string),
hasTarget(?relSujPass, ?npSuj),
hasDescendent(?npSuj,?tReu),
Réu(?tReu) ->  Absolvição(?fcl)

# ABS03 - ABSOLVIÇÃO (voz ativa)
#           Frase: A sentença julgou parcialmente procedente [a ação penal|a denúncia|...]
... para absolver
Absolver(?tAbsolv),
hasParent(?tAbsolv, ?np),
has_cat(?np, "np"^^string),
isSourceOf(?np, ?relP),
has_label(?relP, "P"^^string),
hasTarget(?relP, ?tAbsolv),
hasParent(?np, ?pp),
has_cat(?pp, "pp"^^string),
isSourceOf(?pp, ?relDP),
has_label(?relDP, "DP"^^string),
hasTarget(?relDP, ?np),
isSourceOf(?pp, ?relH),
has_label(?relH, "H"^^string),
hasTarget(?relH, ?prp),
has_lemma(?prp, "para"^^string),
hasAncestral(?pp, ?fcl),
```

```
has_cat(?fcl, "fcl"^^string),
isSourceOf(?fcl,?relS),
has_label(?relS, "S"^^string),
hasTarget(?relS, ?npS),
has_cat(?npS, "np"^^string),
isSourceOf(?npS,?relDN),
has_label(?relDN,"DN"^^string),
hasTarget(?relDN, ?artd),
has_lemma(?artd, "o"^^string),
isSourceOf(?npS, ?relHS),
has_label(?relHS, "H"^^string),
hasTarget(?relHS,?sntca),
has_lemma(?sntca, "sentença"^^string),
isSourceOf(?fcl,?relPfcl),
has_label(?relPfcl, "P"^^string),
hasTarget(?relPfcl, ?tJulgar),
Julgar(?vJulgar),
isSourceOf(?fcl, ?relCo),
has_label(?relCo, "Co"^^string),
hasTarget(?relCo, ?adjp),
has_cat(?adjp, "adjp"^^string),
isSourceOf(?adjp, ?relDA),
has_label(?relDA, "DA"^^string),
hasTarget(?relDA, ?tAdv),
has_lemma(?tAdv,"parcialmente"^^string),
isSourceOf(?adjp, ?relHadjp),
has_label(?relHadjp,"H"^^string),
hasTarget(?relHadjp,?tAdj),
has_lemma(?tAdj,"procedente"^^string)
-> Absolvição(?pp)

# ABS04 - ABSOLVIçÃO (voz ativa)
#           Frase: A sentença absolveu
#           Princípio: o agente do verbo "absolver" é "sentença"
Absolver(?verbo),
has_pos(?verbo, "v-fin"^^string),
hasParent(?verbo, ?fcl),
isSourceOf(?fcl, ?relation),
has_label(?relVAux, "S"^^string),
hasTarget(?relVAux, ?np),
hasChild(?np,?tSent)
has_lemma(?tSent,"sentença"^^string)
 -> Absolvição(?fcl)

# DENÚNCIA (voz ativa)
Denunciar(?verbo),
hasParent(?verbo, ?fcl),
isSourceOf(?fcl, ?relation),
```

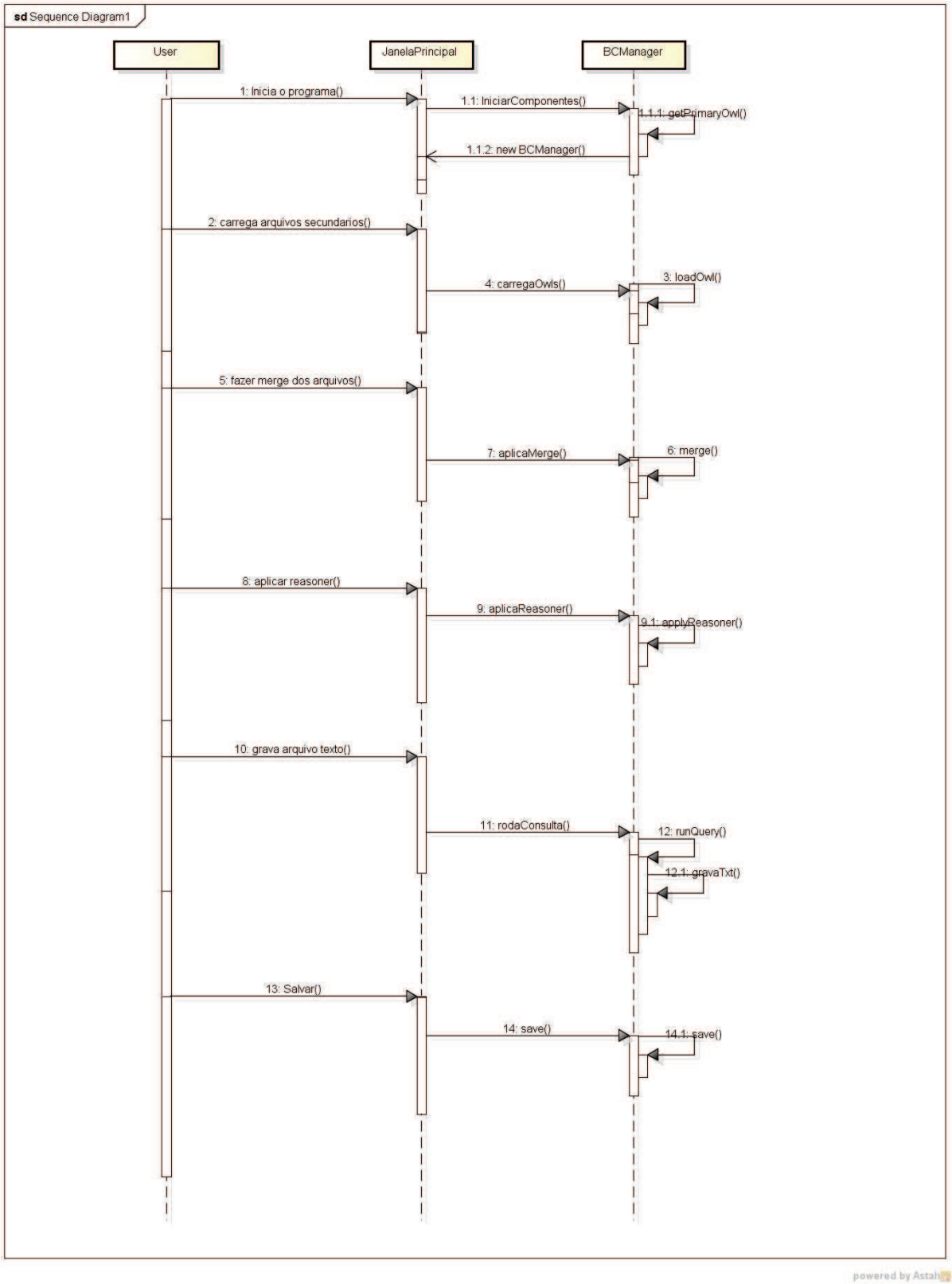
```
has_label(?relVAux, "S"^^string),
hasTarget(?relVAux, ?np),
hasChild(?np,?mp)
Ministério_Público(?mp)
 -> Denúncia(?fcl)

# DENÚNCIA (voz passiva - Réu e MP)
Denunciar(?verbo),
hasParent(?verbo, ?vpDenuncPass),
isSourceOf(?vpDenuncPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
has_pos(?tVAux, "v-fin"^^string),
has_extra(?tVAux, "fmc aux"^^string),
hasParent(?vpDenuncPass, ?fcl),
isSourceOf(?fcl, ?relSujPass),
has_label(?relSujPass, "S"^^string),
hasTarget(?relSujPass, ?npSuj),
hasDescendent(?npSuj,?tReu),
Réu(?tReu),
isSourceOf(?fcl, ?relAgPass),
has_label(?relAgPass, "fApass"^^string),
hasTarget(?relAgPass, ?ppMP),
hasDescendent(?ppMP, ?tMP),
Ministério_Público(?tMP) ->
Denúncia(?fcl)

# DENÚNCIA (voz passiva - Réu e MP v2)
Denunciar(?verbo),
hasParent(?verbo, ?vpDenuncPass),
isSourceOf(?vpDenuncPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
has_pos(?tVAux, "v-fin"^^string),
has_extra(?tVAux, "fmc aux"^^string),
hasParent(?vpDenuncPass, ?fcl),
isSourceOf(?fcl, ?rel),
has_label(?rel, "S"^^string),
hasTarget(?rel, ?npSuj),
hasDescendent(?npSuj,?tReu),
Réu(?tReu),
has_label(?rel, "fApass"^^string),
hasTarget(?rel, ?ppMP),
hasDescendent(?ppMP, ?tMP),
Ministério_Público(?tMP) ->
Denúncia(?fcl)
```

```
# DENÚNCIA (voz passiva - Réu)
Denunciar(?verbo),
hasParent(?verbo, ?vpDenuncPass),
isSourceOf(?vpDenuncPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
has_pos(?tVAux, "v-fin"^^string),
has_extra(?tVAux, "fmc aux"^^string),
hasParent(?vpDenuncPass, ?fcl),
isSourceOf(?fcl, ?relSujPass),
relSujPass, "S"^^string),
hasTarget(?relSujPass, ?npSuj),
hasDescendent(?npSuj,?tReu),
Réu(?tReu) ->
Denúncia(?fcl)

# DENÚNCIA (voz passiva - MP)
Denunciar(?verbo),
hasParent(?verbo, ?vpDenuncPass),
isSourceOf(?vpDenuncPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
has_pos(?tVAux, "v-fin"^^string),
hasParent(?vpDenuncPass, ?fcl),
isSourceOf(?fcl, ?relAgPass),
has_label(?relAgPass, "fApass"^^string),
hasTarget(?relAgPass, ?ppMP),
hasDescendent(?ppMP, ?tMP),
Ministério_Público(?tMP) ->
Denúncia(?fcl)

# CONDENAÇÃO (voz ativa)
Condenar(?tVerbo),
hasParent(?tVerbo, ?fcl),
isSourceOf(?fcl, ?relOd),
has_label(?relOd,"Od"^^string),
hasTarget(?relOd, ?npReu),
hasDescendent(?npReu,?tReu),
Réu(?tReu) -> Condenação(?fcl)

# CONDENAÇÃO (voz passiva - Réu)
Condenar(?verbo),
hasParent(?verbo, ?vpCondenPass),
isSourceOf(?vpCondenPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
```

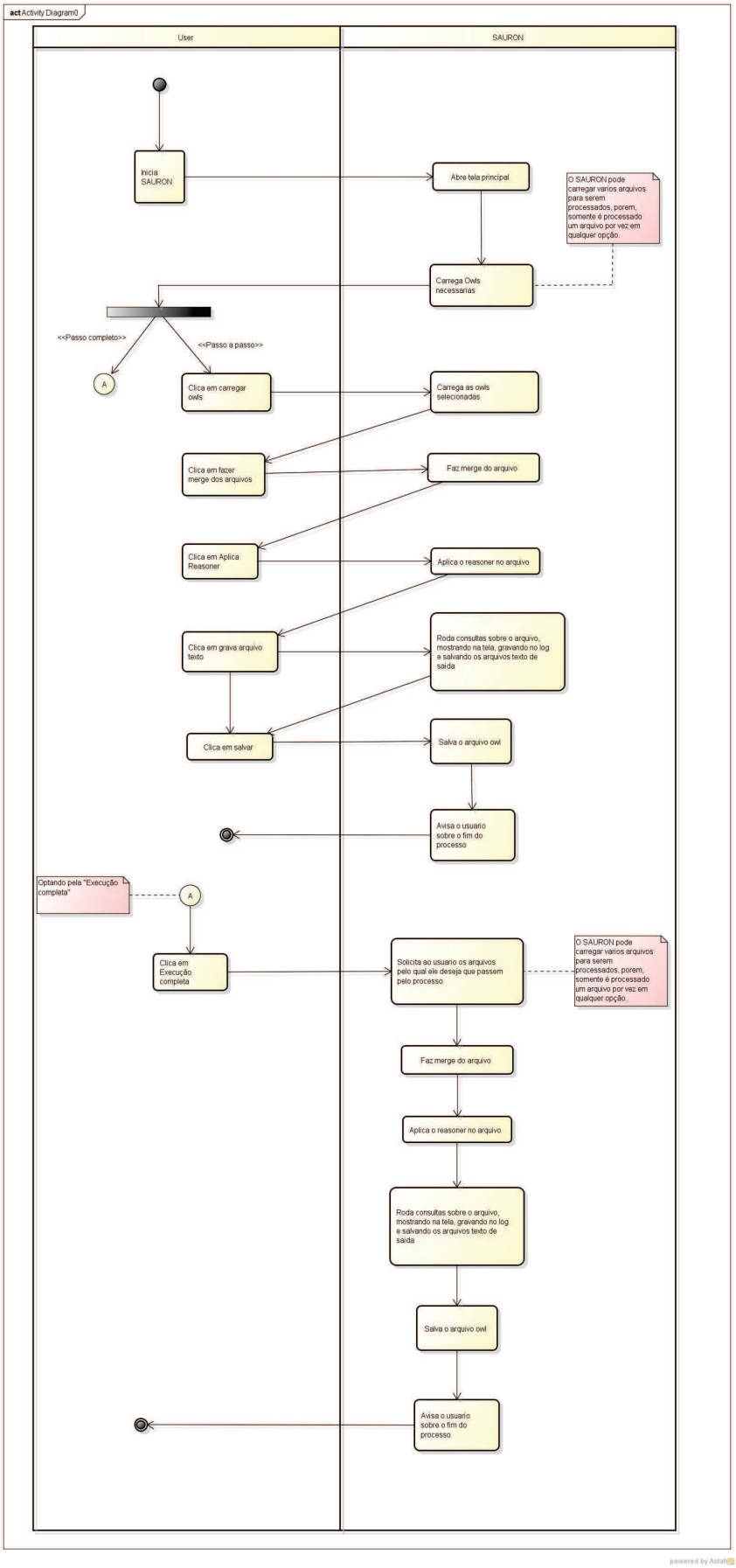
```
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
has_pos(?tVAux, "v-fin"^^string),
has_extra(?tVAux, "fmc aux"^^string),
hasParent(?vpCondenPass, ?nt),
isSourceOf(?nt, ?relSujPass),
has_label(relSujPas, "S"^^string),
hasTarget(?relSujPass, ?npSuj),
hasDescendent(?npSuj,?tReu),
Réu(?tReu) ->
Condenação(?nt)

# INT01 - INTERRROGATÓRIO (forma nominal)
#           Frase: "interrogatório do réu"
Terminal(?t),
has_lemma(?t, "interrogatório"^^string),
hasParent(?t, ?np),
isSourceOf(?np, ?relPP),
hasTarget(?relPP, ?pp),
has_cat(?pp, "pp"^^string),
isSourceOf(?pp,?relNP),
has_label(?relNP, "DP"^^string),
hasTarget(?relNP, ?np2),
hasChild(?np2, ?tReu),
Réu(?tReu) -> Interrogatório(?np)

# INT02 - INTERRROGATÓRIO (voz passiva)
#           Frase: "réu ... foi interrogado" / "o réu ... tendo sido interrogado"
Interrogar(?verbo),
hasParent(?verbo, ?vpPass),
isSourceOf(?vpPass, ?relVAux),
has_label(?relVAux, "Vaux"^^string),
hasTarget(?relVAux, ?tVAux),
has_lemma(?tVAux, "ser"^^string),
hasAncestral(?vpPass,?fcl),
#has_cat(?fcl,"fcl"^^string), ===>> verbo pode não ser finitivo, sintagma idem
isSourceOf(?fcl,?relS),
has_label(?relS, "S"^^string),
hasTarget(?relS, ?ntSuj),
hasDescendent(?ntSuj,?tReu),
Réu(?tReu) -> Interrogatório(?fcl)

# INT03 - INTERRROGATÓRIO (forma nominal)
#           Frase: "o réu ... foi ... submetido a interrogatório
Terminal(?t),
has_lemma(?t, "interrogatório"^^string),
hasParent(?t, ?pp),
has_cat(?pp, "pp"^^string),
```

```
isSourceOf(?pp,?relDP),
has_label(?relDP, "DP"^^string),
hasTarget(?relDP,?t),
#isSourceOf(?pp,?relH),
#has_label(?relH, "H"^^string),
#hasTarget(?relH,?prp),
#has_pos(?prp,"prp"^^string),
hasParent(?pp, ?icl),
isSourceOf(?icl,?relP),
has_label(?relP, "P"^^string),
hasTarget(?relP,?verbo),
has_lemma(?verbo, "submeter"^^string),
has_pos(?verbo, "v-pcp"^^string),
hasAncestral(?icl,?fcl),
has_cat(?fcl,"fcl"^^string),
isSourceOf(?fcl,?relS),
has_label(?relS, "S"^^string),
hasTarget(?relS, ?ntSuj),
hasDescendent(?ntSuj,?tReu),
Réu(?tReu) -> Interrogatório(?icl)
```

APÊNDICE C DIAGRAMAS DO SISTEMA SAURON



8


act Activity Diagram0
User
SAURON
Inicia SAURON
Abre tela principal
O SAURON pode carregar varios arquivos para serem processados, porem, somente é processado um arquivo por vez em qualquer opção.
Carrega Owls necessarias
<<Passo completo>>
<<Passo a passo>>
A
Clica em carregar owls
Carrega as owls selecionadas
Clica em fazer merge dos arquivos
Faz merge do arquivo
Clica em Aplica Reasoner
Aplica o reasoner no arquivo
Clica em grava arquivo texto
Roda consultas sobre o arquivo, mostrando na tela, gravando no log e salvando os arquivos texto de saida
Clica em salvar
Salva o arquivo owl
Avisa o usuario sobre o fim do processo
Optando pela "Execução completa"
A
Clica em Execução completa
Solicita ao usuario os arquivos pelo qual ele deseja que passem pelo processo
O SAURON pode carregar varios arquivos para serem processados, porem, somente é processado um arquivo por vez em qualquer opção.
Faz merge do arquivo
Aplica o reasoner no arquivo
Roda consultas sobre o arquivo, mostrando na tela, gravando no log e salvando os arquivos texto de saida
Salva o arquivo owl
Avisa o usuario sobre o fim do processo
powered by Astah