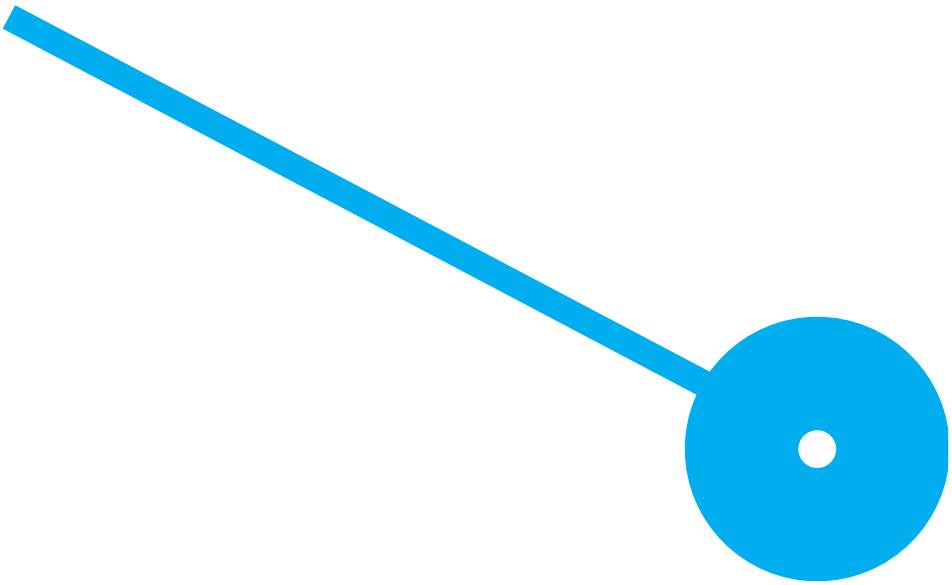
ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO
POLITÉCNICO
DO PORTO

MESTRADO
ENGENHARIA INFORMÁTICA

Recuperação Inteligente de Informação Legal, através de princípios Semânticos e Inteligência Artificial

Mário Jorge Mendes Leite

10/2023



ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO
POLITÉCNICO
DO PORTO

MESTRADO
ENGENHARIA INFORMÁTICA

Recuperação Inteligente de Informação Legal, através de princípios Semânticos e Inteligência Artificial

Mário Jorge Mendes Leite
8170573

Orientador(es)

Prof. Doutor Cristóvão Dinis Polido Sousa
Prof. Doutor Bruno Moisés Teixeira de Oliveira

Dissertação apresentada para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Informática pela Escola Superior de Tecnologia e Gestão do Instituto Politécnico do Porto.

10/2023

Declaração de Integridade

Eu, Mário Jorge Mendes Leite, estudante nº 8170573, do Mestrado de Engenharia Informática da Escola Superior de Tecnologia e Gestão do Instituto Politécnico do Porto, declaro que não fiz plágio nem auto-plágio, pelo que o trabalho intitulado "Recuperação Inteligente de Informação Legal, através de princípios Semânticos e Inteligência Artificial" é original e da minha autoria, não tendo sido usado previamente para qualquer outro fim. Mais declaro que todas as fontes usadas estão citadas, no texto e na bibliografia final, segundo as regras de referenciação adotadas na instituição.

Agradecimentos

A elaboração da presente tese não seria possível sem o apoio de várias pessoas.

Em especial, ao meu orientador, Professor Doutor Cristóvão Sousa, e ao meu coorientador, Professor Doutor Bruno Oliveira, por me terem orientado, incentivado e apoiado continuamente ao longo de todo este projeto.

À minha família, principalmente padrinho e avó, que sempre estiveram do meu lado e sempre me apoiaram incondicionalmente ao longo deste jornada. Sem eles, nada disto seria possível.

A todos os docentes da Escola Superior de Tecnologia e Gestão pelos conhecimentos que me transmitiram ao longo do curso.

Ao CIICESI, Centro de Inovação e Investigação em Ciências Empresariais e Sistemas de Informação, pela bolsa de investigação e à FCT, Fundação para a Ciência e a Tecnologia, pelo financiamento que permitiu o desenvolvimento deste trabalho.

O meu muito obrigado!

Resumo

Qualquer domínio técnico, seja o domínio jurídico, fiscal, político, social, é caracterizado por um discurso altamente especializado usando uma terminologia própria. Para além disso, o conhecimento encontra-se ainda pouco estruturado e disperso por vários recursos textuais. Estas características apresentam um desafio à representação digital do conhecimento, de modo a assegurar a recuperação inteligente de informação, adequada ao contexto.

Neste trabalho é proposto o desenvolvimento de um sistema de recuperação de informação tendo por base uma ontologia do domínio do direito fiscal desenvolvida numa dupla perspetiva, ou seja, capturando e representando a experiência dos especialistas do domínio, complementada com técnicas de processamento de linguagem natural.

Além disso, tendo por base a ontologia, aplicou-se princípios de *machine learning* para previsão e anotação semântica de *"Named Entities"* para acrescentar meta-informação aos documentos legais. É importante que esta ontologia suporte a incorporação de outros vocabulários já existentes de modo a criar um sistema de recuperação inteligente de informação através de pesquisas facetadas com possibilidade de consultar conteúdos semelhantes ou relevantes para um determinado contexto.

Palavras-chave: Domínio Jurídico, Inteligência artificial, Processamento de Linguagem Natural, Sistema de Recuperação de Informação

Abstract

Any technical field, be it legal, fiscal, political or social, is characterised by a highly specialised discourse using its own terminology. In addition, knowledge is still poorly structured and scattered across various textual resources. These characteristics present a challenge to the digital representation of knowledge, in order to ensure the intelligent retrieval of information, appropriate to the context.

This work proposes the development of an information retrieval system based on an ontology of the tax law domain developed from a dual perspective, i.e. capturing and representing the experience of domain experts, complemented with natural language processing techniques.

In addition, based on the ontology, machine learning principles were applied to the prediction and semantic annotation of Named Entities to add meta-information to legal documents. It is important that this ontology supports the incorporation of other existing vocabularies in order to create an intelligent information retrieval system through faceted searches with the possibility of consulting similar or relevant content for a given context.

Keywords: Legal Domain, Artificial Intelligence, Natural Language Processing, Information Retrieval System.

Conteúdo

Lista de Figuras

Exemplos de Código

Siglas e Abreviaturas

CMS *Content Management System*

DRE Diário da República Eletrónico

DGSI Direção-Geral dos Serviços de Informática

DSR *Design Science Research*

ELI *European Legislation Identifier*

HTTP *Hypertext Transfer Protocol*

IA Inteligência Artificial

IR *Information Retrieval*

JSONL *JavaScript Object Notation Lines*

KaaS *Knowledge-as-a-Service*

KOS *Knowledge Organization System*

LKIF *Legal Knowledge Interchange Format*

LPG *Labeled Property Graph*

NER *Named Entity Recognition*

NLP *Natural Language Processing*

OIF O Informador Fiscal

OWL *Web Ontology Language*

POS *Part-Of-Speech Tagging*

RDF *Resource Description Framework*

UI *User Interface*

UE União Europeia

URI *Uniform Resource Identifier*

W3C *World Wide Web Consortium*

Capítulo 1

Introdução

Atualmente, vivemos na era do *Big Data* e da digitalização. A quantidade de dados e documentos gerada diariamente pelas organizações cresce a um ritmo exponencial [1]. No entanto, o simples facto de uma organização possuir grandes quantidades de dados por si só, não é suficiente para suportar o processo de tomada de decisão [2]. As organizações começaram a perceber que garantir apenas o acesso aos dados não é suficiente. Os dados deixam de ser apenas importantes para gerir as atividades operacionais, e começam a ser utilizados para apoiar os procedimentos analíticos. Por este motivo, vários domínios começaram a preocupar-se com a qualidade dos dados o que implica assegurar uma estrutura de dados bem definida e, por ventura, enriquecer semanticamente os dados com o objetivo de criar e extrair valor dos mesmos [3].

No entanto, estruturar e enriquecer domínios mais complexos, como o domínio jurídico, social ou político, requer profundo conhecimento do mesmos. Estes domínios são caracterizados por um discurso altamente especializado usando uma terminologia e estilo próprios na codificação textual dos temas que lhe estão subjacentes. Para além disso, os recursos que constituem a base de conhecimento destes domínios estão, tipicamente, pouco ou nada estruturados e dispersos por vários recursos textuais [4]. Estas características apresentam um desafio à representação digital do conhecimento, de modo a assegurar a recuperação inteligente de informação, adequada a um determinado contexto.

O domínio do direito, mais concretamente do direito fiscal, é constituído por um grande volume de informação não-estruturada de elevada tecnicidade, categorizada por códigos tributários (Estatuto de Benefícios Fiscais, Código do IVA, Código do IRS, etc.) e decretos-lei. A

interpretação e recuperação de informações de modo inteligível, ou seja, informação contextualizada face à base de conhecimento existente, é ainda mais difícil devido às características deste domínio. As diversas fontes de informação, que compõem este domínio técnico, são caracterizadas por um elevado grau de interdependência, onde as remissões inter e intra-textuais e as atualizações de conteúdo são frequentes [4]. Por exemplo, as leis fiscais são constantemente revistas e atualizadas, através de decretos-lei ou leis, para se adaptarem às mudanças na economia e na sociedade.

Por outro lado, são documentos que a nível de estruturação apresentam características peculiares, desafiam as técnicas normalmente utilizadas para processamento de linguagem natural, pela variedade de pontuações, estrutura linguística e sintaxe que apresentam [5]. É comum fazer-se uso de abreviaturas para identificar outros documentos (art.º 98º da Lei Geral Tributária), que permitem consolidar ou compreender melhor o determinado contexto de um documento, ou entidades jurídicas (ANPD - Autoridade Nacional de Proteção de Dados). No entanto, estas características dificultam significativamente a estruturação e o processamento de informação jurídica.

Além disso, os termos do domínio jurídico não apresentam um carácter objetivo, pelo contrário, existe uma grande ambiguidade entre termos que designam conceitos [6]. O termo ”dedução”, é um exemplo da existência desta ambiguidade, em que, se releva necessário analisar o contexto para identificar o significado do termo. Onde, no mesmo documento, o termo pode significar a dedução de um valor ou o processo de raciocínio. Não será ao acaso que a versão digital do Diário da República Eletrónico (DRE)[1], passou a incluir, recentemente um ”Lexionário”[2].

Para os especialistas do domínio a ambiguidade dos termos deixa de ser um problema graças à sua experiência e facilidade de compreender o contexto perante um determinado documento [7]. Os significados e usos da lei são difíceis de capturar ou prever, pois os dados jurídicos estão em constante evolução e dependem do contexto, muitas vezes é necessário um determinado grau de literacia para identificar o significado apropriado para uma lei num determinado contexto. Porém, este é um aspeto comprometedor na interpretação de texto pelas ferramentas e algoritmos de processamento de linguagem natural [8].

Em Portugal, houve vários esforços para oferecer conteúdo relacionado ao direito de forma digital. Algumas dessas iniciativas são públicas, como o DRE e a Direção-Geral dos Serviços

[1] https://dre.pt/dre/home
[2] https://dre.pt/dre/lexionario

de Informática (DGSI)[3], enquanto outras são privadas, como o *Lexit*[4]. Este último é o mais avançado em termos de digitalização e oferece uma plataforma digital para apoiar os utilizadores disponibilizando conteúdo útil para suas atividades de prática jurídica. Além da vasta biblioteca legislativa, disponibilizam anotações (sumários e/ou comentários de teor prático e científico, que incluem referências a excertos da doutrina e jurisprudências relevantes, oferecendo um *corpus*[5] técnico de valor acrescentado) incorporadas nos códigos legais, artigos de opinião e casos práticos. Todo o conteúdo fornecido é gerido por uma equipa de cerca de 80 colaboradores especializados em um ou vários domínios, oferecendo aos utilizadores da plataforma digital contexto e orientação para interpretações das leis. Esta abordagem colaborativa e focada na partilha de conhecimento tem garantido ao O Informador Fiscal (OIF) o estatuto de um dos principais intervenientes do mercado na área, garantindo o cumprimento dos deveres fiscais e reduzindo os encargos aos seus utilizadores. Apesar do *Lexit* se apresentar como uma das plataforma mais avançada em termos de digitalização, a informação é fundamentalmente não estruturada ou estruturada de acordo com normas próprias dificultado a tarefa de recuperação de informação.

O processo de recuperação de informação é essencialmente baseado em termos-chave ou num conjunto de critérios previamente estabelecidos. Não existem pesquisas facetadas[6] e os atuais mecanismos de recuperação de informação apresentam resultados excessivos e pouco relevantes, muitas das vezes desenquadrados e completamente descontextualizados.

Além disso, apenas é possível efetuar consultas com enfoque apenas num tipo de recurso e os resultados não são contextualizados com o restante corpus, ou seja, não existe um relacionamento entre diferentes recursos textuais, uma vez que o *Lexit* trata os diferentes recursos informacionais de forma independente e não correlacionada. Concretamente, os *datasets* com conteúdo sobre códigos e diplomas, e as operações de pesquisa e recuperação de informação que lhe estão subjacentes, não incluem os restantes *datasets* com conteúdo acerca de casos práticos, jurisprudência e/ou outros recursos informacionais. Este isolamento resulta na formação ”silos de informação”, o que dificulta a utilização e o relacionamento dos diversos recursos textuais. Na Figura 1 alínea a) é retratado este problema e, na 1 alínea b), é representado a situação ideal que relacione os diversos recursos textuais.

De um modo geral, existem barreiras tecnológicas e arquiteturais à recuperação de informação

[3] http://www.dgsi.pt/
[4] https://informador.pt/legislacao/lexit/
[5] Conjunto de documentos.
[6] Tipo de pesquisa que permite navegar e filtrar resultados em categorias ou aspetos específicos.

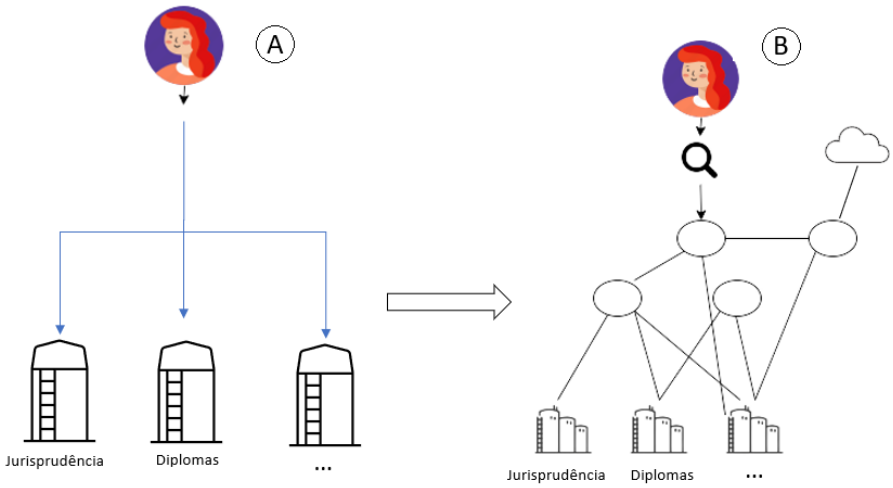


Figura 1: Problema dos silos de informação

contextualizada. Estes métodos básicos de pesquisa e o isolamento da informação, tornam o processo de recuperação de informação demorado e descontextualizado, resultando em informações desalinhadas ou que não respondem completamente ao contexto de pesquisa. Este aspeto é um fator crítico de sucesso considerando as características das fontes de informação (informação textual não-estruturada) que compõem este domínio. Somando as características específicas do domínio legal, as práticas de recuperação de informação carecem de uma abordagem mais "inteligente" e estruturada. A existência de informação excessiva, desenquadrada e desorganizada, pode condicionar todo o processo de exploração de dados, podendo levar a decisões erradas caso seja indevidamente interpretada. Logo, pretende-se facilitar a interação e orientação do utilizador com a informação disponível, apresentando informação útil e contextualizada.

Para isso, é necessário recorrer a processos de identificação de informação relevante, do inglês *Information Retrieval* (IR), considerando a existência de dados tipicamente não estruturados e em grandes quantidades. É importante garantir que o utilizador comum, num contexto como a pesquisa de informação, encontre a informação mais relevante para aquele cenário, evitando que seja levado a interpretações equivocadas e confusões no processo de tomada de decisão [9].

Portanto, neste contexto surge a seguinte questão de investigação como principal desafio da presente dissertação: **Como estruturar e organizar digitalmente documentos jurídicos, de modo a suportar atividades de pesquisa e recuperação de informação baseadas no contexto, contribuindo para oferecer conteúdo útil no âmbito das tarefas relacionadas com o desenvolvimento de práticas jurídicas.** Para responder a esta questão de investigação,

seguiu-se uma abordagem híbrida, combinando princípios semânticos com princípios de inteligência artificial, com a ambição de dotar de maior inteligência o processo de recuperação de informação.

1.1 Objetivos

Esta dissertação tem com principal propósito o desenvolvimento de um sistema de recuperação de informação tendo por base o desenvolvimento de uma ontologia do domínio do direito fiscal desenvolvida numa dupla perspetiva, ou seja, capturando e representando o conhecimento e a experiência dos especialistas do domínio, complementada com técnicas de processamento de linguagem natural, do inglês *Natural Language Processing* (NLP). Numa segunda fase, tendo por base a ontologia, aplicam-se princípios de *machine learning* para previsão e anotação semântica de *Named Entities* para uma caracterização conceptual dos diplomas legais através da adição de meta-informação aos documentos. É importante que esta ontologia suporte a incorporação de outros vocabulários já existentes de modo a criar um sistema de recuperação de informação enquadrado com as necessidades dos utilizadores, permitindo a criação de pesquisas facetadas com possibilidade de consultar conteúdos semelhantes ou relevantes para um determinado contexto. Em suma, pretende-se:

1. Implementar e validar uma ontologia do domínio fiscal, tendo como principal vantagem o envolvimento dos especialistas do domínio, para suportar e garantir a eficiência no processo de recuperação inteligente de informação legal;
2. Criar um *dataset* para identificação de entidades, do inglês - *Named Entity Recognition* (NER), para o domínio jurídico português;
3. Identificar técnicas de *machine learning* e *frameworks* que permitam extrair informações relevantes dos documentos jurídicos em português;
4. Desenvolver um conjunto de serviços/componentes, seguindo uma abordagem *Knowledge-as-a-Service* (KaaS) que permita entregar ao utilizador conhecimento (dados contextualizados) ao invés de dados ou informação, em que cada serviço/componente na arquitetura tem um propósito bem definido que permita suportar e garantir a recuperação eficiente de informação.

1.2 Metodologia

A elaboração do plano de trabalhos obedeceu a uma lógica técnico-científica fundamentada na metodologia *Design Science Research* (DSR) [10], com uma ênfase significativa na participação contínua dos especialistas do domínio ao longo de todo o ciclo de vida do projeto, procurando compreender as suas necessidades e perspetivas para, de seguida, idealizar uma solução.

A DSR é uma metodologia aplicada na especificação de soluções e desenho de artefactos, cuja a utilidade dos artefactos é um dos principais atributos de qualidade. Aplicam-se princípios de engenharia ao rigor científico na procura de atingir resultados exequíveis e mensuráveis, alinhados com as necessidades inerentes ao problema inicial [11]. O resultado da aplicação da metodologia DSR é sempre um artefacto (processo, ferramenta, tecnologia, etc.) para atingir um determinado objetivo [12]. A Figura 2 apresenta, de forma simples, as diferentes fases do projeto tendo por base esta metodologia, nomeadamente: (1) Estudo do Domínio; (2) Análise do Problema; (3) Desenvolvimento dos Artefactos; (4) Integração e Validação dos Artefactos.

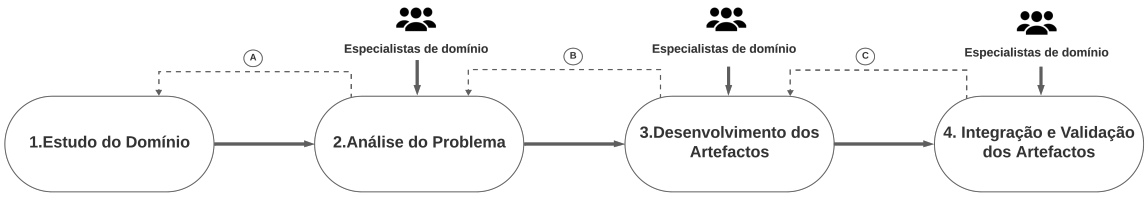


Figura 2: Fases do projeto

Apesar do uso de uma abordagem incremental e interativa cuja progressão é controlada através de ciclos de relevância, avaliação e rigor (representados pelas letras A, B, C na Figura, respetivamente), cada uma destas fases contribuí para o resultado da seguinte e vice-versa. Uma vez que, o resultado de uma fase posterior pode exigir uma alteração numa fase anterior.

A primeira fase (1) do projeto consistiu no estudo do domínio, compreender o modelo de negócio, as necessidades e problemas comuns do domínio e uma revisão da literatura. O resultado desta tarefa permitiu adquirir um conhecimento mais profundo do domínio, dos principais desafios e problemas e dos projetos que têm vindo a ser publicados neste domínio.

Numa segunda fase (2), através de sucessivas reuniões e do envolvimento dos especialistas do domínio, consistiu por analisar o problema em concreto, analisar as fontes de dados e um estudo das diferentes ferramentas que possam vir a ser utilizadas na implementação da

solução a ser desenvolvida. O resultado desta fase consistiu na caracterização das fontes de dados, na definição dos objetivos e na especificação dos requisitos dos artefactos a ser desenvolvidos.

A terceira fase (3), consistiu no desenvolvimento de um protótipo da solução, idealizado e implementado nesta fase. Sempre que necessário foi revisitada a fase anterior para clarificar o problema ou avaliar se as decisões seguidas no desenvolvimento dos artefactos se enquadram com os requisitos.

A última fase (4), consistiu na avaliação da solução e respetivas melhorias, considerando os aspetos **qualitativos** (utilidade dos artefactos) recorrendo a uma abordagem **baseada em caso de estudo**, tendo por base o *feedback* dos especialistas do domínio.

1.3 Estrutura do Documento

A dissertação encontra-se dividida da seguinte forma:

- **Capítulo 1 -** *Introdução*: Neste capítulo é apresentado o contexto, o problema em mãos, os objetivos que se pretende atingir com o trabalho e a metodologia seguida. Por fim, é apresentada a estrutura da dissertação;
- **Capítulo 2 -** *Recuperação de informação*: Neste capítulo são apresentados conceitos teóricos importantes para compreender o âmbito da dissertação e uma revisão da literatura;
- **Capítulo 3 -** *Modelo de representação de conhecimento para recuperação de informação*: Neste capítulo é apresentada a abordagem, a conceptualização do domínio e a arquitetura conceptual.

  Capítulo 4 - *Especificação dos artefactos*: Neste capítulo são apresentados os artefactos desenvolvidos ao longo do projeto;
- **Capítulo 5 -** *Caso de estudo*: Neste capítulo é apresentado o caso de estudo em que se insere esta dissertação, são apresentados e discutidos os resultados qualitativos tendo por base a utilidade dos artefactos no caso de estudo apresentado;
- **Capítulo 6 -** *Conclusões e Trabalho Futuro*: Neste capítulo é feita uma reflexão crítica e descrito o trabalho futuro.

Capítulo 2

Recuperação de Informação

Hoje em dia, a *Internet* é a forma mais fácil e rápida de procurar informação. Com o aumento constantes dos dados presentes nas plataformas digitais, os processos de recuperação de informação assumem um papel importante e essencial [13]. São responsáveis por identificar, recolher, armazenar, interpretar e por fim, garantir que a informação apresentada aos utilizadores é a mais adequada e precisa em determinado contexto.

Neste capítulo são apresentadas duas áreas que contribuem de diversas formas para o desenvolvimento de mecanismos de recuperação eficientes e enquadrados com as necessidades dos utilizadores. Na secção 2.1 são apresentadas as contribuições da área das ciências da informação, os conceitos e as diversas estruturas propostas nesta área para organizar e explorar conhecimento. Na secção 2.2, são apresentadas as contribuições da área da Inteligência Artificial (IA) e as técnicas e ferramentas de extração de informação propostas na literatura. Por fim, são apresentados trabalhos relacionados na área do domínio jurídico enquadrados com a áreas tecnológicas referenciadas.

2.1 Ciências da Informação

Entre os diversos ramos de especialidade da Ciência da Informação, está a área de organização e representação do conhecimento, que se dedica ao estudo das teorias, metodologias e estruturas para organizar e representar conhecimento. [14]. A utilização de estruturas de organização de conhecimento é fundamental para suportar o processo de recuperação de

informação, permitindo a representação de características do domínio (e respetivo contexto) de forma a suportar a utilização de dados de forma inequívoca e expressiva [15].

A área das ciência da informação procura oferecer o "*HOW*" para organizar e estruturar o conhecimento de forma mais eficiente, através da definição dos processos e de estruturas de organização de informação. É frequente o relacionamento com outras áreas, principalmente com a evolução tecnologias e ferramentas na IA nas últimas décadas, para processamento dos dados [16]. A evolução de IA permite colocar em prática os artefactos de estruturação e organização especificados no âmbito na disciplina de ciências de informação.

Com o surgimento da *web* semântica, uma extensão da atual *web*, que tem como principal objetivo adicionar significado semântico à informação disponível *online*, de forma a que esta seja facilmente compreensível e processável tanto por humanos como por computadores [17]. Diversos modelos de representação de conhecimento, conhecidos na literatura por *Knowledge Organization System* (KOS) (na área das ciências da informação), têm vindo a ser propostas e avaliadas por diversos autores [18] [19].

Os KOS variam em formato e representação, mas partilham o mesmo objetivo de organizar e gerar conhecimento, através da definição de significados semânticos, propriedades, conceitos e relacionamentos entre os dados [15]. Esta dissertação incide sobre duas formas de representação de conhecimento no domínio jurídico: taxonomias e ontologias.

2.1.1 Taxonomias

As taxonomias são mencionadas na literatura como ferramentas essenciais na organização da informação [20]. As taxonomias são um sistema de classificação que define uma estrutura hierárquica lógica (classificação) para um conjunto de conceitos de um determinado domínio. Este tipo de estrutura facilita o processo de recuperação e partilha de informação no contexto de uma organização [21].

Muitas vezes, utilizámos este tipo de estruturas sem a noção da sua existência. O exemplo, a Figura 3, apresenta parte de uma taxonomia, que poderia ser utilizada para organizar dispositivos eletrónicos numa loja de informática. No topo da hierarquia são colocados os conceitos mais gerais (como: famílias de produtos) e na parte mais inferior surgem os produtos [22].

A utilização deste tipo de estruturas aumenta a expressividades do dados, fornecendo-se um

contexto hierárquico e permite uma exploração mais sofisticada dos dados. Em contextos como a gestão de produtos eletrónicos, este enriquecimento dos dados traduz-se na melhoria da experiência do utilizador ao interagir com o catálogo de produtos, permitindo uma navegação mais intuitiva sobre a lista de produtos. Mesmo com esta organização relativamente modesta dos dados utilizando hierarquias taxonómicas, existem benefícios imediatos em termos do conhecimento que pode ser obtido com base nas hierarquias [22]. Por exemplo, podemos afirmar que *wireless earphones* é um resultado válido para alguém que procura dispositivos de áudio porque a taxonomia afirma que é uma subcategoria de *earhphones* que por sua vez é uma subcategoria de áudio.

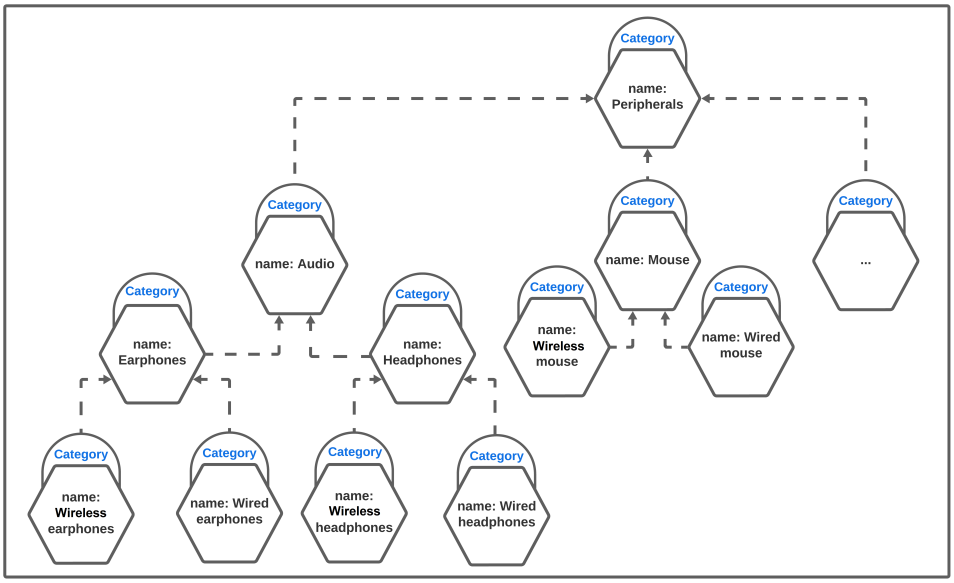


Figura 3: Representação em grafo de uma taxonomia para dispositivos eletrónicos

2.1.2 Ontologias

Por definição, uma ontologia é uma representação específica e formal de uma conceptualização partilhada de um domínio [23]. É específica porque define claramente conceitos, relacionamentos, instâncias e axiomas relevantes para o domínio. É formal, porque é legível e interpretável por máquinas. É partilhada, porque o seu conteúdo é consensual dentro de uma determinada comunidade/domínio. As ontologias representam uma conceptualização de um domínio [24].

Em comparação com as taxonomias que apenas definem uma relação hierárquica, as ontologias são estruturas mais robustas e complexas, representando outro tipo de relações entre conceitos do domínio. Uma ontologia cria um modelo semântico para um domínio e permite

compreender de forma mais profunda e precisa sobre o ”mundo” de um domínio em específico. Através da definição destes elementos é possível processar instâncias e inferir novo conhecimento, através de mecanismos de ”raciocínio” que operam sobre as estruturas de organização de conhecimento representadas sob a forma de ontologia.

Estas características são extremamente valiosas na definição de sistemas de recuperação de informação, onde a interpretação dos dados é importante para suportar a eficiência e precisão no processo de recuperação de informação. Segundo os autores de [25]. Algumas razões que podem levar a definição de uma ontologia são:

- Partilhar conhecimento comum entre as pessoas ou agentes de *software*;
- Permitir a reutilização de conhecimento do domínio;
- Tornar explícitos os pressupostos de domínio;
- Separar o conhecimento do domínio do conhecimento operacional;
- Analisar o conhecimento do domínio.

A construção de uma ontologia envolve, inicialmente, a definição do domínio e âmbito em que se insere. Apesar de, nem todas as ontologias serem definidas com base na mesma metodologia ou linguagem, a maioria possui os mesmos elementos básicos [26]:

- **Classes**: Representam conjuntos ou categorias de objetos. São utilizadas para agrupar indivíduos que partilham características semelhantes.
- **Indivíduo**: Instâncias específicas de uma classe.
- **Propriedades**: Descrevem relacionamentos entre classes e indivíduos. Existem dois tipos de propriedades: propriedades de objeto (relacionam indivíduos a outros indivíduos ou classes) e propriedades de dados (relacionam indivíduos a valores de dados).
- **Axiomas**: Entidades que descrevem relacionamentos entre classes e entres classes e indivíduos. São factos considerados como sempre verdadeiros.

A Figura 4, baseada em [22], representa um excerto de uma ontologia que poderia ser utilizada para ajudar a orientar um cliente no processo de decisão de compra. Neste exemplo

existem duas classes (*mobile phone* e *IOS*), três indivíduos/instâncias (*iPhone15 mini, iPhone 15, iPhone 15 pro*, um axioma (*subclass*, descreve que *IOS* é uma subclasse de *mobile phone*), uma propriedade de objeto (*upsell*) e apesar de não representadas na figura, as especificações de cada telemóvel seriam consideradas propriedades de dados.

Seguindo a estrutura da ontologia é possível explorar as categorias do domínio verticalmente (hierarquicamente) e horizontalmente. Por exemplo, é possível afirmar que o *iPhone 15* é um resultado de pesquisa válido para um cliente que procura um telemóvel, porque é um dispositivo *IOS* e a ontologia afirma que o *IOS* é uma subcategoria de telemóvel. E ainda, a partir da relação semântica da relação *Upsell* definida na ontologia, podemos concluir que um *iPhone 12 Pro* pode ser recomendado aos clientes que possuem um *iPhone 15 Pro* [22].

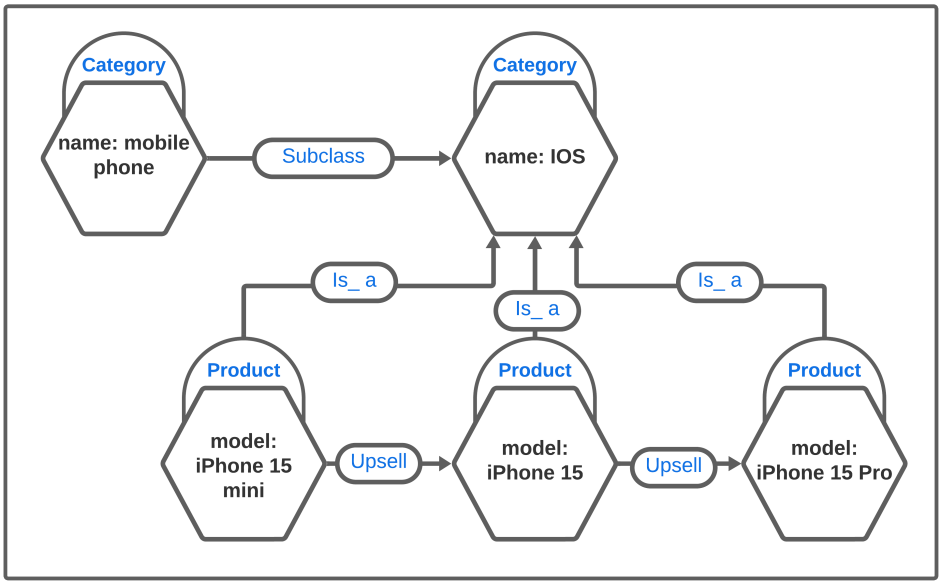


Figura 4: Representação de parte de uma ontologia para dispositivos eletrónicos

De forma a possibilitar a interpretação de ontologias, de forma a que sejam facilmente compreendidas e processadas por máquinas, diversas linguagens têm vindo a ser desenvolvidas e propostas pela comunidade para definir ontologias. Os autores, em [27] [28] apresentam e comparam algumas dessas linguagens.

A *Web Ontology Language* (OWL), uma extensão da linguagem *Resource Description Framework* (RDF)[1], é uma linguagem declarativa utilizada para definir ontologias na *semantic web* e recomendada pela W3C. Tem como principal objetivo acrescentar valor semântico à representação de conceitos através de um vocabulário mais extenso. Esta linguagem pode

[1]Padrão da *World Wide Web Consortium* (W3C) para representar informações e dados na forma de triplas sujeito-predicado-objeto.

ser analisada por *softwares* para verificar a consistência dos dados ou para extrair conhecimento implícito[2]. As ontologias definidas em OWL, podem ser publicadas na W3C e podem referir-se ou ser referidas a partir de outras ontologias OWL [29].

As ontologias, baseadas em *OWL*, desempenham um papel importante no processo de recuperação de informação digital porque permitem enriquecer semanticamente os dados, garantir a sua consistência, extrair conhecimento implícito e permitem ainda promover a interoperabilidade entre diferentes sistemas.

2.1.3 Grafos de conhecimento

Os grafos de conhecimento, são uma estrutura de organização de dados, que através de factos descrevem as entidades, eventos e inter-relacionamentos num formato que é compreendido tanto por humanos como por máquinas [22]. Abordagens orientadas a grafos têm vindo a ganhar popularidade nos últimos anos nas diversas áreas por serem bastantes eficientes e flexíveis para analisar e modelar domínios mais complexos [30].

Tipicamente, este tipo de estruturas de organização de dados, utilizam um princípio de organização de informação que define uma camada semântica de meta-dados (como uma ontologia ou uma taxonomia), que impõe regras de estrutura e interpretação dos dados [22]. Esta estrutura revela-se particularmente útil num contexto de recuperação de informação porque permite construir mecanismos de recuperação de informação mais eficientes e enquandrados com as necessidades dos utilizadores [31].

Os dois modelos de dados baseados em grafos mais utilizados para representação de informação são: RDF e o *Labeled Property Graph* (LPG) [32]. Ambos os modelos têm na sua base a estrutura básica dos grafos, onde os nós, relacionamentos e propriedades desempenham papéis essenciais.

Os grafos de propriedades são os mais comuns, em que os nós representam as entidades, tem uma *label* associada que define o tipo de nó e podem ter zero ou mais propriedades (no formato chave-valor). Os relacionamentos definem como se relacionam os nós, também possuem uma *label* associada que caracterizam o tipo de relacionamento, podem ou não ter propriedades e são direcionados (com um inicio e um fim, em que o fim pode ser o próprio

[2]Conhecimento não explicitamente declarado, mas que pode ser deduzido com base nas relações e regras definidas na ontologia.

nó) [22].

Nos grafos RDF a representação de informação assume um formato específico e segue padrões bem definidos para garantir a interoperabilidade e a semântica dos dados. Uma ”afirmação” é chamada de triplo, e possuí a forma de sujeito-predicado-objeto. Em que o sujeito representa a entidade, o objeto pode ser uma entidade ou um valor e o predicado representa a relação entre o sujeito e o objeto. Esta é uma característica que distingue um grafo RDF em relação ao LPG, que é mais flexível em termos de estrutura e não impõe um esquema rigoroso.

Embora sejam semelhantes, os LPG são muito compactos em termos de tamanho e diminuí consideravelmente o tamanho do grafo de conhecimento em comparação com o RDF. Os LPG permitem poupar espaço de armazenamento, consultar informações de forma mais direita (através de *queries*) e simplificam a visualização das informações. Em contraste, os grafos RDF facilitam a definição de *queries* mais complexas e oferecem capacidades de raciocínio quando suportados por uma ontologia. [33]

Diversos autores, como [32] [34], propõe abordagens em que utilizam os dois tipos de grafos. Os autores tiram partido das vantagens dos LPG, para representar instâncias dos dados, e estruturam e enriquecem semanticamente os dados através de ontologias ou taxonomias com o objetivo de criar uma representação mais completa e enriquecida do conhecimento, permitindo uma compreensão semântica das relações e significados subjacentes dos dados.

2.1.4 Vocabulários para estruturação do domínio jurídico

Nas últimas décadas, diversos vocabulários têm vindo a ser propostos para estruturar o domínio legal com o objetivo de facilitar o acesso, a partilha e a interoperabilidade da informação jurídica. Entre estes vocabulários destacam-se duas abordagens, desenvolvidas em contextos de projetos europeus, o *Legal Knowledge Interchange Format* (LKIF)[3] *Core* e o *European Legislation Identifier* (ELI)[4].

O LKIF *Core* é uma ontologia, desenvolvida em conformidade com os padrões da *web* semântica, para o domínio jurídico que faz parte de uma arquitetura genérica para sistemas de conhecimento jurídico, conhecida como LKIF. O LKIF tem dois objetivos principais: permitir a tradução entre bases de conhecimento jurídico escritas em diferentes formatos e formalismos

[3] https://github.com/RinkeHoekstra/lkif-core
[4] https://eur-lex.europa.eu/eli-register/about.html

de representação e, em segundo lugar, definir um formato de representação de conhecimento que faz parte de uma arquitetura mais ampla para o desenvolvimento de sistemas de conhecimento jurídico [35].

No contexto deste objetivo, o LKIF *Core* desempenha um papel significativo, concentra-se em representar apenas os conceitos fundamentais e nas relações no contexto do domínio jurídico. Essa ontologia é dividida em vários módulos que cobrem diferentes aspetos do domínio legal e conceitos relacionados com o senso comum [35].

O ELI tem como objetivo incorporar a legislação na *web*, facilitando o acesso, partilha e interoperabilidade de informação jurídica na União Europeia (UE). Propõe a definição de um identificador único e estruturado para cada ato legislativo Europeu (regulamentos, diretivas, decisões, etc), conhecido como *Uniform Resource Identifier* (URI), com base em componentes comuns e a descrição dos seus meta-dados com base em uma ontologia [36].

Esta ontologia do ELI define as propriedades dos dados necessários para descrever a legislação (como: país de origem, tipo de ato, ano em que foi criado, etc) e *links* (outros URI) para relacionar atos legislativos. Isso é especialmente importante em um contexto em que vários sistemas e países estão envolvidos na interpretação e implementação da legislação da UE. O *ELI* está intimamente relacionado com as novas arquiteturas de sistemas de informação jurídica, baseadas em grafos de conhecimento jurídico [36]. O DRE integra o ELI para identificar e padronizar documentos legais, como leis, regulamentos e decretos, assegurando a interoperabilidade e a transparência das informações legais.

Os vocabulários LKIF *Core* e o ELI desempenham papéis importantes na estruturação do domínio jurídico, permitindo uma representação semântica consistente e a identificação única de atos legislativos. Neste contexto, a ontologia fornece apoio de várias formas: fornece mecanismos de inferência, define os termos na ontologia para facilitar a organização de conhecimentos e facilita a interoperabilidade de conhecimento entre várias bases de conhecimentos.

2.2 Inteligência artificial

Nos últimos anos, a utilização de técnicas inovadoras para processamento, modelação e utilização dos dados tornou-se amplamente popular, especialmente com a evolução das técnicas de IA [37]. Diversas técnicas e algoritmos de *machine-learning* têm vindo a ser propostos para

suportar o desenvolvimento de sistemas em domínios mais complexos [38].

Destacam-se duas técnicas de IA particularmente úteis para o processo de recuperação e extração de informação aplicadas ao domínio jurídico: técnicas de processamento de linguagem natural (NLP, secção 2.2.1) e de identificação de entidades (NER, secção 2.2.2). No contexto de recuperação de informação, estas técnicas permitem compreender e extrair *insights* valiosos a partir de grandes volumes de texto não estruturado.

2.2.1 Técnicas de NLP

O processamento de linguagem natural (NLP) é uma subárea da IA que estuda a compreensão e geração automática de linguagem natural, verbal ou escrita, por máquinas [39]. Esta técnica permite extrair informação de documentos (estruturados ou não-estruturados) e pode ser aplicada em diversos contextos, como por exemplo: resumir textos [40], transformar diálogo em texto e vice-versa [41], tradução automática de texto [42], análise de sentimentos [43] ou deteção de notícias falsas [44].

Segundo [45], as tarefas mais simples em NLP incluem tarefas que lidam com os aspetos fundamenteis do processamento de texto (compressão e a análise semântica do texto), sendo a base para construir as ferramentas/aplicações de alto nível mencionadas anteriormente. A eficácia no processo de extração de informação depende destas fases de pré-processamento, tais como:

- ***Tokenization***: processo de segmentar o texto em palavras únicas ou *tokens*, que permitem identificar pontuação, dígitos, entre outros. A Figura 5 demonstra o exemplo do processo de *tokenization* numa frase;

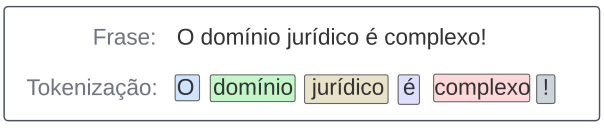


Figura 5: *Tokenization*

- ***Lemmatization***: processo que consiste na análise morfológica das palavras para identificar a sua forma mais básica. No exemplo da Figura 5, existe um único *lemma*: ”ser” (é)

- **POS**: processo que consiste em identificar a classe gramatical[5] de cada palavra numa frase ou texto, podem ser verbos, adjetivos, substantivos, etc. A Figura 6, demonstra um exemplo deste processo;

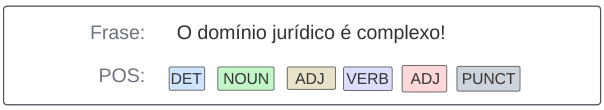


Figura 6: POS

- ***Dependency Parsing***: processo que analisa a estrutura gramatical das frases e identifica as relações sintáticas entre as palavras. A Figura 7, apresenta um exemplo deste processo.

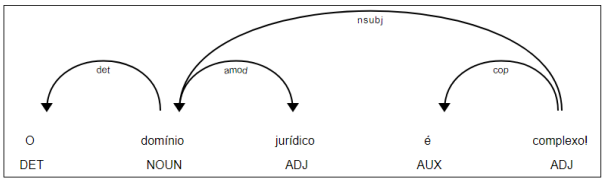


Figura 7: *Dependecy Parsing*

2.2.2 *Named-entity recognition*

NER é uma das principais aplicações de NLP, que tem como principal objetivo identificar entidades presentes no texto de um documento (estruturado ou não-estruturado), classificando-o dentro de um conjunto de entidades (ou *labels)* pré-definidas para um determinado contexto. Numa abordagem NER, uma entidade consiste num termo presente no texto que pode ser composto por um ou mais *tokens* (palavras), enquanto, o tipo de entidade difere de acordo com cada abordagem específica. [46].

Num contexto de identificação de entidades, as mais comuns são: pessoas, locais, datas, organizações, etc. A Figura 8, apresenta um exemplo de uma possível abordagem para identificação de entidades num excerto de texto. Foram identificadas três entidades: uma organi-zação (Autoridade Tributária e Aduaneira), uma localização (Lisboa) e uma data (15 de abril de Abril de 2023).

As técnicas de identificação de entidades diferem em dois tipo de abordagens: uma abordagem baseada em técnicas *hand-conded* e um abordagem baseada em técnicas de *machine*

[5]https://universaldependencies.org/u/pos/

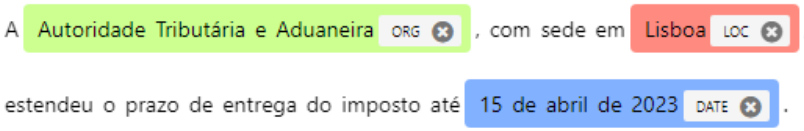


Figura 8: Exemplo do processo de identificação de entidades.

learning [47]. A primeira abordagem consiste numa abordagem manual que tem como objetivo identificar entidades usando regras, padrões gramaticais ou dicionários pré-definidos. A segunda abordagem, mais automatizada em comparação com a abordagem anterior, tem por base técnicas de *machine learning* [6]. Contudo, é necessário treinar um modelo com um conjunto de dados previamente anotados. Em [47], concluiu-se que as abordagens híbridas (abordagens que utilizam ambas as técnicas) apresentam os melhores resultados.

2.3 Recuperação de informação jurídica

A criação, gestão, partilha e utilização eficaz de conhecimento são vitais para as organizações não só a nível intraorganizacional mas também a nível interorganizacional [48]. No domínio jurídico, pela sua complexidade, responder a questões legais é uma tarefa desafiante e suscetível a erros, uma vez que, os documentos estão sujeitos a diferentes interpretações e pode ser necessário consultar diferentes recursos textuais [7].

Com o aumento da quantidade de informação digital, os utilizadores necessitam de sistemas ”inteligentes”, ou seja, de acordo com as suas necessidades, para suportar o processo de tomada de decisão. Contudo, é importante garantir a eficiência, precisão e interoperabilidade destes sistemas [49]. O conhecimento extraído dos documentos jurídicos suportará o processo de tomada de decisão, isto significa que, o tratamento inadequado dos dados jurídicos pode levar a induzir os utilizadores em erro e a tomar decisões desastrosas [7].

Diversos trabalhos têm vindo a ser publicados, no âmbito do domínio jurídico, com o objetivo de propor mecanismos de recuperação de informação, recorrendo a abordagens e estruturas apresentadas anteriormente nesta dissertação. Alguns destes trabalhos seguem processos e abordagens definidos na área da ciência da informação, recorrendo a estruturas de informação para estruturar o conhecimento(KOS). Em [50], os autores introduziram, *JudO*, uma biblioteca

[6]Uma área da IA que estuda a capacidade de aprendizagem de máquinas, ao longo do tempo, através da experiência, a partir de dados ou dos resultados das suas ações, sem serem explicitamente programadas.

de ontologia OWL2 no domínio jurídico, para modelar a informação relacionada com jurisprudências. É suportada por uma base de conhecimento, para agilizar e facilitar o processo de argumentação durante as decisões judiciais. Em [51], os autores apresentam um modelo formal de normas jurídicas, modeladas em OWL, utilizado para desenvolver sistemas especializados para a elaboração semiautomática, recuperação semântica e navegação na legislação.

Ainda em [52], os autores descrevem como as ontologias legais podem oferecer soluções interessantes para formalizar conhecimento jurídico e propõem uma abordagem construtiva para a definição de novos componentes que melhorem os sistemas de suporte à decisão jurídica, localmente ou na forma de serviços semânticos na *web*.

Outros trabalhos, seguem abordagens que tem por base técnicas de IA. Como em [53], os autores procuraram extrair informação de um conjunto de documentos, provenientes de fontes heterogéneas, relacionados com investigações criminais realizadas em Portugal. Com o objetivo de facilitar o processo de tomada de decisão pelos oficiais de justiça na análise e compreensão destes documentos. Para isso, desenvolveram uma *pipeline* de processamento de linguagem natural como uma tarefa de NER, treinada com dois corpus diferentes, que fosse capaz de identificar entidades relevantes neste contexto (organizações, pessoas, ou locais). Apesar dos resultados promissores, os autores concluíram que o modelo desenvolvido necessita de ser treinado com um corpus com uma qualidade superior ao usado para treinar o modelo inicial.

Na literatura estão presentes ainda abordagens híbridas, abordagens que têm por base estruturas de informação e técnicas de IA. Em [54], os autores implementaram uma sistema eficiente de pesquisas, que tem por base tarefas de NLP para identificar entidades e relacionamentos. Utilizaram um grafo para representar a informação extraída e, por fim, através de uma pesquisa semântica é possível realizar pesquisas sobre o grafo. O utilizador, como *input*, introduz uma palavra-chave e recebe como *output* os julgamentos semanticamente relacionados. Ainda, em [55] os autores propõe uma ontologia para fundamentar relações entre textos jurídicos relacionados com o direito e o procedimento criminal nos Estados Unidos. São usadas técnicas de NLP para extrair informação dos documentos jurídicos, mapeando-a para regras específicas da ontologia do domínio jurídico para suportar uma ferramenta que dá respostas a questões legais.

Em suma, estes trabalhos evidenciam as dificuldades e a importância da otimização de processo de recuperação de informação legal. Nos vários trabalhos apresentados, a utilização

de ontologias é um ingrediente comum para a representação de conhecimento. Através da definição de uma camada semântica, proporcionam uma descrição formal do domínio através de um conjunto de relacionamentos entre os vários conceitos de domínio.

Combinando esta abordagem com as diferentes técnicas de NLP com o objetivo de extrair informação e enriquecer semanticamente os dados, para melhorar a eficiência e precisão no processo de recuperação de informação legal. Para além disso, a informação enriquecida pode ser processada recorrendo não só a processos de exploração de dados que utilizam várias técnicas para a indexação e análise dados, mas também por motores de inferência que utilizam mecanismos de relacionamento de factos e regras para suportar validação das regras de domínio e, eventualmente, para a descoberta de novos factos.

Capítulo 3

Modelo de representação de conhecimento para recuperação de informação

O processo de resolução de questões fiscais, implica a sua contextualização jurídica através de um articulado baseado num *corpus* legal. Contudo, as constantes alterações legislativas, que podem, em algumas circunstâncias, alterar a perceção dos contextos jurídicos, aliado a carência de recursos humanos de elevada especialização e de instrumentos com elevado nível de resposta às exigências dos especialistas, tornam este processo desafiador. O tratamento preciso de questões fiscais e a sua disponibilização de forma organizada e adequada às necessidades dos utilizadores, pode criar um valor acrescentado à empresa, mas sobretudo desempenhar um papel fundamental na prevenção de conflitos com as autoridades tributárias.

Neste capitulo, é descrita a abordagem técnico cientifica seguida (secção 3.1). Em seguida, são apresentadas as duas *stacks* conceptuais: *stack* semântica (secção 3.2) e a *stack* de IA (secção 3.3), por fim, é apresentada a arquitetura conceptual da solução desenvolvida (secção 3.4).

3.1 Abordagem técnico-científica

Apesar do processo de anotação do corpus poder ser considerado uma prática epistêmica[1] e cujos comentários incluem referências e excertos da doutrina e jurisprudência relevantes e úteis para responder a questões jurídicas, o sistema de organização do conhecimento inerente a este processo, está longe de estar sistematizado e padronizado, com vista à pesquisa e recuperação facilitada da informação que lhe está subjacente.

Para além disso, o processo de tratamento de questões jurídicas resulta, frequentemente, na consulta de dados de múltiplas fontes de informação, onde cada documento está sujeito a atualizações regulares e constantes, o que complica substancialmente todo o processo. Neste contexto, o processo de gestão do conhecimento assume-se como crítico, desempenhando um papel fundamental no desempenho organizacional e uma ferramenta importante para a garantir a competitividade [56].

Neste sentido, é proposto um modelo de representação de conhecimento para recuperação de informação baseado numa abordagem com enfoque em duas disciplinas complementares, incorporando princípios semânticos e princípios de IA. Caracterizamos esta abordagem como sócio-semântica, em que os artefactos desenvolvidos procuram representar o domínio e a experiência dos seus especialistas através do desenvolvimento de um processo colaborativo orientado por princípios de engenharia do conhecimento, representando os conceitos do domínio através de uma rede semântica.

Por outro lado, são utilizadas técnicas de IA, mais concretamente técnicas de NLP e NER, para processamento e extração de informação dos documentos jurídicos. Estes mecanismos ou técnicas permitem instanciar o modelo semântico (exemplo: uma ontologia) com os principais termos e conceitos jurídicos, criando um grafo de conhecimento com os meta-dados relevantes do *corpus* jurídico. Deste modo, é possível desenvolver mecanismos de recuperação de informação mais ricos quer do ponto de vista da forma como do processo. A Tabela 1, resume as contribuições de cada área para o modelo em específico e para a resolução do problema em geral.

[1] Atividade ou processo que adiciona conhecimento

Semântica	IA
• Processo de conceptualização • Sistema de organização de conhecimento (ontologia) • Abordagem centrada nas pessoas e na iteração social	• Processamento do corpus • Extração terminológica usando NLP • Identificação de entidades específicas do domínio através de modelos NER

Tabela 1: Contribuições de cada área

Na especificação do modelo, considera-se que os algoritmos de IA serão mais eficazes se tiverem alicerçados num modelo de raciocínio com base numa conceptualização colaborativa do conhecimento, permitindo definir fronteiras e limitar a classificação automática de termos a documentos. Na prática significa que os resultados dos algoritmos e técnicas de IA são controlados, não permitindo que tirem novos pressupostos para tentar preencher lacunas nos dados que futuramente condicionem o processo de tomada de decisão jurídica.

A Figura 9 representa o modelo de atuação dos artefactos semânticos e dos artefactos de IA. Os artefactos semânticos, acomodam os resultados dos algoritmos de IA, através de uma *pipeline* que irá alinhar os resultados desses algoritmos com o sistema de organização do conhecimento, implementado sob a forma de grafo do conhecimento. Através de mecanismos de *reasoning*, é possível oferecer interfaces mais sofisticados de pesquisa, navegação e recuperação de informação sobre o corpus existente.

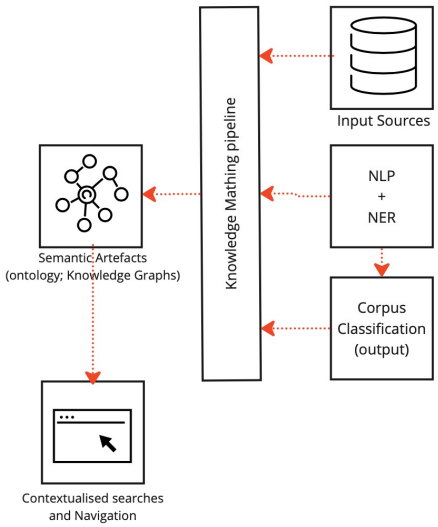


Figura 9: Alinhamento dos artefactos semânticos e artefactos de IA

3.2 Stack semântica

Na abordagem distinguem-se duas *stacks* conceptuais: i) a *stack* sócio-semântica, conforme representado na Figura 10, e; ii) a *stack* associada à abordagem focada na inteligência artificial, discutida na secção 2.2 e representada na Figura 13.

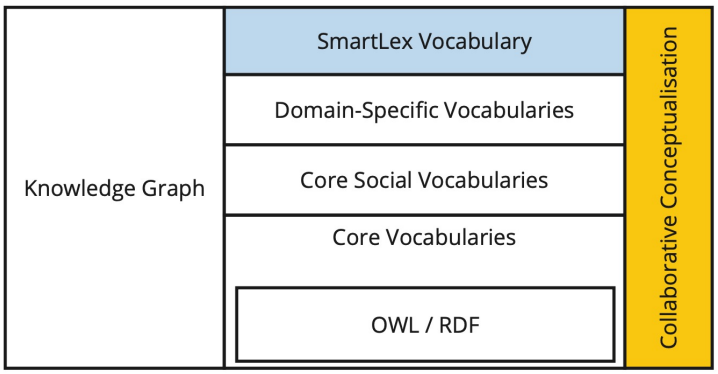


Figura 10: Stack Socio-Semântica

O pilar da abordagem sócio-semântica reside na construção de uma conceptualização do domínio. Esta conceptualização, na prática, resultará num grafo de conhecimento cuja função é persistir as instâncias (artigos, diplomas, termos, conceitos jurídicos, ...). Em termos formais a conceptualização, é traduzida numa ontologia em OWL. A ontologia, para além de acomodar o vocabulário específico do problema, deverá integrar outros vocabulários específicos do domínio jurídico como o ELI e LKIF. No sentido de elevar a sofisticação do processo de recuperação de informação, a ontologia pretende integrar com vocabulários sociais (exemplo: FOAF). O objetivo é enriquecer os resultados de pesquisa de acordo com as relações sociais entre os utilizadores. Estas relações são definidas com base na subscrição de conteúdos de determinados autores. No entanto, para que seja possível beneficiar desta integração, a plataforma *Lexit* terá de implementar a funcionalidade de subscrição de conteúdo. Adicionalmente, o modelo semântico final deverá integrar o vocabulário SIOC, permitindo a articulação entre os *posts* do *Lexit* (implementado em *wordpress*), os subscritores e os conceitos do domínio.

3.2.1 Conceptualização inicial do domínio

O processo de conceptualização do domínio pretende definir uma visão conceptual dos conceitos e relações conceptuais no domínio legal, em particular do domínio do direito fiscal. O modelo inclui, não apenas conceitos relacionados com o tipo de conteúdo presente no *corpus*

legal, mas também conceitos relacionados com o conteúdo específico da plataforma digital *Lexit*. O resultado deste processo colaborativo corresponde a uma estrutura que permite definir uma representação formal e sistemática dos conceitos, relações e propriedades dentro do domínio. Esta estrutura evolui no sentido da sua formalização em OWL, constituindo o elemento base da base de conhecimento.

O processo de conceptualização colaborativo, ilustrado na Figura 11, contempla várias tarefas, nas quais os especialistas do domínio e os engenheiros de conhecimento trabalham em conjunto com o objetivo de criar um modelo conceptual do domínio jurídico, representando as principais entidades e respetivos relacionamentos.

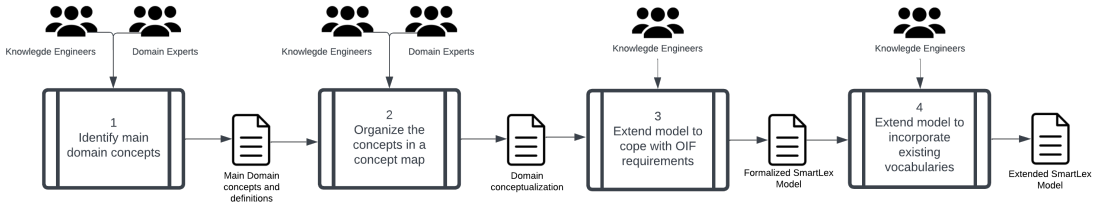


Figura 11: Processo conceptualização colaborativo

A tarefa inicial (1) começa com a identificação dos principais conceitos-chave do domínio jurídico, tendo por base a experiência dos especialistas do domínio e os conteúdos produzidos pelo OIF. Na segunda tarefa (2) são organizados os conceitos num modelo conceptual para validação e revisão dos conceitos numa representação mais visual.

Na terceira tarefa (3) o modelo é enriquecido e mapeado para os conceitos na plataforma do OIF. Por último, tarefa (4), o modelo é expandido para suportar vocabulários jurídicos já existentes, nomeadamente o ELI e o LKIF. O trabalho descrito neste documento, inicia-se a partir da fase 3, existindo já uma conceptualização inicial do domínio, conforme representado da Figura 12

3.3 Stack de Inteligência Artificial

A *stack* de IA, representada na Figura 13, por sua vez, tem o seu foco nos mecanismos de processamento, análise e classificação de conteúdo existente no *corpus*. Neste sentido, a unidade mais básica de informação desta *stack* é texto, que poderá ser agregado, constituído um corpus de informação específico de determinado domínio. A partir de um corpus é possível realizar atividades mais ou menos sofisticadas sobre o texto, no sentido de conhecer o mesmo,

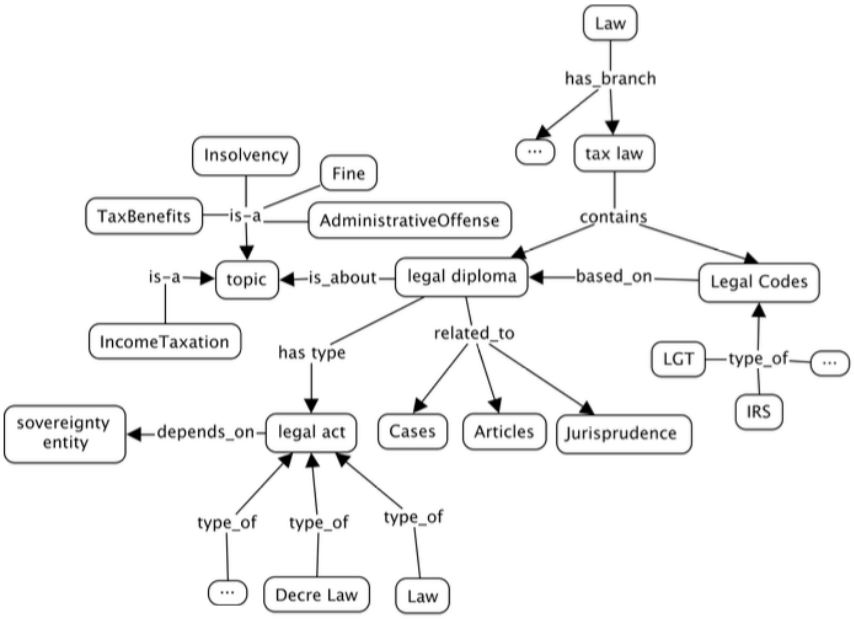


Figura 12: Conceptualização Inicial

quer de um ponto de vista da sua composição sintática e organização estrutural, quer de um ponto de vista mais analítico, permitindo a classificação do texto quanto ao seu conteúdo. Se para um processamento mais sintático, se utilizam algoritmos de NLP mais básicos e baseados em métodos estatísticos, já para a abordagem mais analítica será necessário algoritmos mais sofisticados como NER. Os resultados destes algoritmos permitem enriquecer o grafo de conhecimento. O desenvolvimento destes artefactos foi um processo iterativo e incremental procurando afinar os algoritmos de NLP, a descrição e implementação destes técnicas é feita no capítulo 4.

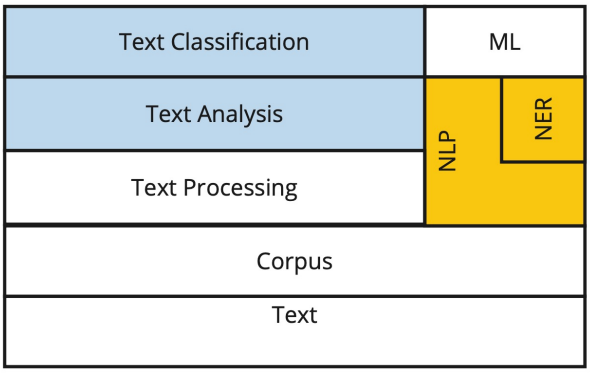


Figura 13: Stack de IA

Face aos avanços da literatura e à especificidade do domínio jurídico, pretende-se encapsular os processos de identificação, tratamento e enriquecimento de informação seguindo um paradigma ”conhecimento como um serviço” (KaaS), em que um fornecedor de serviços de conhecimento, através do seu servidor de conhecimento, responde a questões apresentadas

por consumidores de conhecimento [57]. Os serviços estão distribuídos em várias camadas, cada um com um propósito específico, todos coordenados por um *gateway* encarregue de facilitar a interação e compor diferentes perspetivas com base no conhecimento subjacente.

A solução desenvolvida baseia-se em serviços autónomos que se integram facilmente com os sistemas existentes, permitindo que o OIF compartilhe esses serviços com seus parceiros ou clientes, impulsionando não apenas a escalabilidade da solução, mas também do modelo de negócios como um todo. Estes componentes suportam todo o processo de recuperação ”inteligente” de informação e permitem enriquecer o modelo de informação.

3.4 Arquitetura Conceptual da Solução

O modelo descrito na secção anterior pressupõe o desenvolvimento e orquestração de artefactos semânticos como serviços de IA, uma vez que, por si só, os artefactos/serviços são apenas elementos desconexos que pouco valor acrescentam ao processo de pesquisa e recuperação de informação. O modelo que define o sistema de organização de conhecimento (KOS) e os serviços de IA, é consubstanciado numa arquitetura orientada a serviços, cujo propósito fundamental é orquestrar a forma de como entregar conhecimento útil aos utilizadores.

A Figura 14, apresenta a arquitetura conceptual da solução que pretende mitigar os problemas anteriormente apresentados. Um dos requisitos do OIF era que a solução fosse independente sem a necessidade de alterar em grande parte o atual componente. Neste sentido, foi idealizada uma solução constituída por diferentes serviços ou componentes distribuídos por diferentes camadas, cada uma com um propósito bem definido, mas que juntos resultam num sistema que tem como principal objetivo melhorar a eficiência no processo de recuperação de informação. Nesta arquitetura existem três componentes essenciais:

- **OIF** - Este componente consiste na atual *stack* tecnológica da plataforma de que faz parte o *Lexit*. É a típica aplicação em *wordpress*, baseada em *PHP* e com uma base de dados em *MySQL*. É neste componente que está armazenado todo conteúdo legal (códigos anotados e comentados, legislação, caso-práticos, artigos de opinião, etc) que o OIF disponibiliza para apoiar os seus utilizadores na prática de atividades jurídicas. Neste componente é ”espelhada” a ontologia jurídica, orientada ao domínio fiscal, presente no *metadata broker* para apresentar e relacionar os diferentes tipos de recursos

textuais;

- **KaaS *Ecosystem*** - Este componente suporta todo o processo de recuperação de informação e permite enriquecer o modelo de informação, segue uma abordagem KaaS e procura extrair conhecimento (dados contextualizados) ao invés de apenas dados ou informação dos recursos textuais. É constituído por diferentes serviços distribuídos por diferentes camadas, cada um com um propósito bem definido e que comunicam entre si de forma assíncrona[2]. Estes serviços são responsáveis por realizar diversas tarefas, relacionadas com catalogação semântica[3], como: acrescentar meta-dados (provenientes de fontes internas e externas - outros vocabulários), extrair termos relevantes dos *corpus* dos documentos (processo denominado por: extração terminológica) e aplicar modelos, técnicas e algoritmos de inteligência artificial (técnicas de NER e NLP) que permitam extrair conhecimento;
- ***Metadata Broker*** - Este componente contém toda a lógica de acesso à base de conhecimento e tem como principal função permitir o acesso aos dados estruturados e contextualizados a aplicações internas (OIF) ou externas. Este componente faz uso de uma ontologia, do domínio legal, para gerar e armazenar a representação do conhecimento obtido através das diferentes tarefas no componente KaaS *Ecosystem*. Através deste componente será possível desenvolver *queries* complexas que permitam construir vistas contextualizadas e personalizadas sobre o conteúdo processado, garantindo a contextualização da informação e permitindo a interoperabilidade entre diferentes recursos textuais.

Para além dos componentes, a Figura 14 apresenta as iterações entre cada componente presente na arquitetura:

1. O utilizador interage com o componente OIF para recuperar informação jurídica. Não necessita de conhecer o funcionamento do resto dos componentes;
2. O componente do OIF utiliza uma *interface* do *Metadata Broker* para consultar (realizar *queries)* sobre o conteúdo armazenado neste componente e com base nos resultados, criar as visualizações para os utilizadores. Com este *interface* apenas é possível realizar consultas, não é possível manipular os conteúdos armazenados;

[2]Mecanismo de comunicação não bloqueante, o serviço não espera pela resposta
[3]Processo de organização e classificação de informação de acordo com o seu significado

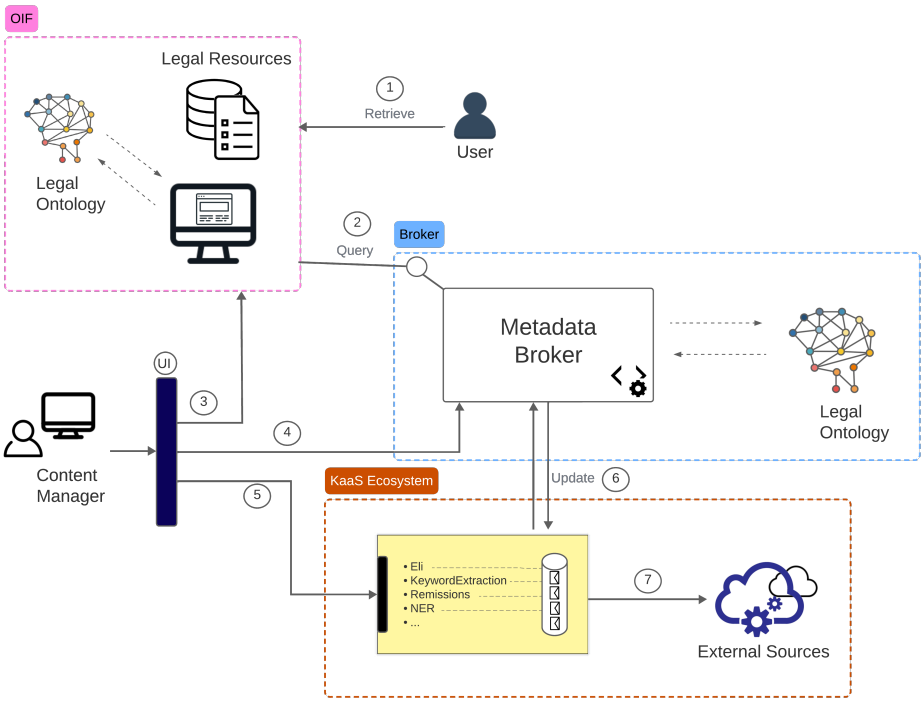


Figura 14: Arquitetura conceptual

3. O responsável por gerir o conteúdo (na Figura14 identificado como *Content Manager*) pode gerir o conteúdo do componente *OIF*, incluindo tarefas como visualizar novos documentos ou documentos que ainda não foram processados na base de conhecimento;

4. Através de uma *User Interface* (UI), o responsável por gerir o conteúdo pode visualizar e manipular os dados presentes no componente do *metadata broker*. É importante garantir a integridade da informação no *metadata broker* para os resultados serem precisos e úteis. Isto inclui tarefas como, gerir os documentos e os relacionamentos no grafo de conhecimento, gerir o *output* dos serviços do KaaS *Ecosystem*, etc.

5. O responsável por gerir o conteúdo pode interagir com o KaaS *Ecosystem* para inserir novos documentos, atualizar documentos, utilizar os serviços, etc;

6. A comunicação entre estes dois componentes é uma das mais importantes do sistema. É através deste mecanismos que o grafo de conhecimento é alimentado e enriquecido com o resultado dos serviços desenvolvidos.

7. Representa comunicações externas com outras fontes de informação para enriquecer a informação na base de conhecimento. Um dos exemplos, é o DRE que disponibiliza diariamente os diplomas com meta-informação (ELI).

Os componentes são todos independentes entre si, tornando o sistema mais disponível, na medida em que, se um componente falhar não compromete os restantes. Para não existir replicação de informação, os documentos encontram-se armazenados apenas no componente do OIF e apenas existe uma referência no *metadata broker* para cada documento processado pelos serviços de conhecimento.

Capítulo 4

Especificação de artefactos

A solução desenvolvida baseia-se num conjunto de serviços autónomos que compõem a solução KaaS, cujo propósito é entregar conhecimento útil aos utilizadores da plataforma, permitindo que o OIF partilhe esses serviços com seus parceiros ou clientes, impulsionando não apenas a escalabilidade da solução, mas também do modelo de negócios como um todo. Este componente suporta todo o processo de recuperação de informação e permite enriquecer o modelo de informação.

Neste capítulo são descritos ao pormenor os artefactos desenvolvidos ao longo do projeto. Na secção 4.1 é apresentado o resultado do processo de conceptualização. Em seguida, na secção 4.2 é descrita a implementação da *pipeline* de NLP, na secção 4.3 a abordagem e o treino do modelo de NER e, por fim, na secção 4.4 é apresentada e descrita a arquitetura final da solução desenvolvida.

4.1 Especificação da conceptualização

Como foi dito anteriormente, o processo de especificação da conceptualização de domínio foi desenvolvida de forma colaborativa, cuja iteração inicial constituiu o desenvolvimento de um modelo conceptual de alto nível. Em iterações posteriores, procurou-se integrar os conceitos do domínio com outros vocabulários existentes. Primeiramente, o foco na integração com a ontologia ELI e LKIF (c.f Figura 15), depois a integração com os vocabulários FOAF e SIOC (c.f. Figura 16).

A formalização do modelo conceptual, com base em princípios SKOS e OWL procura mapear os conceitos do domínio com os principais conceitos do *wordpress*.

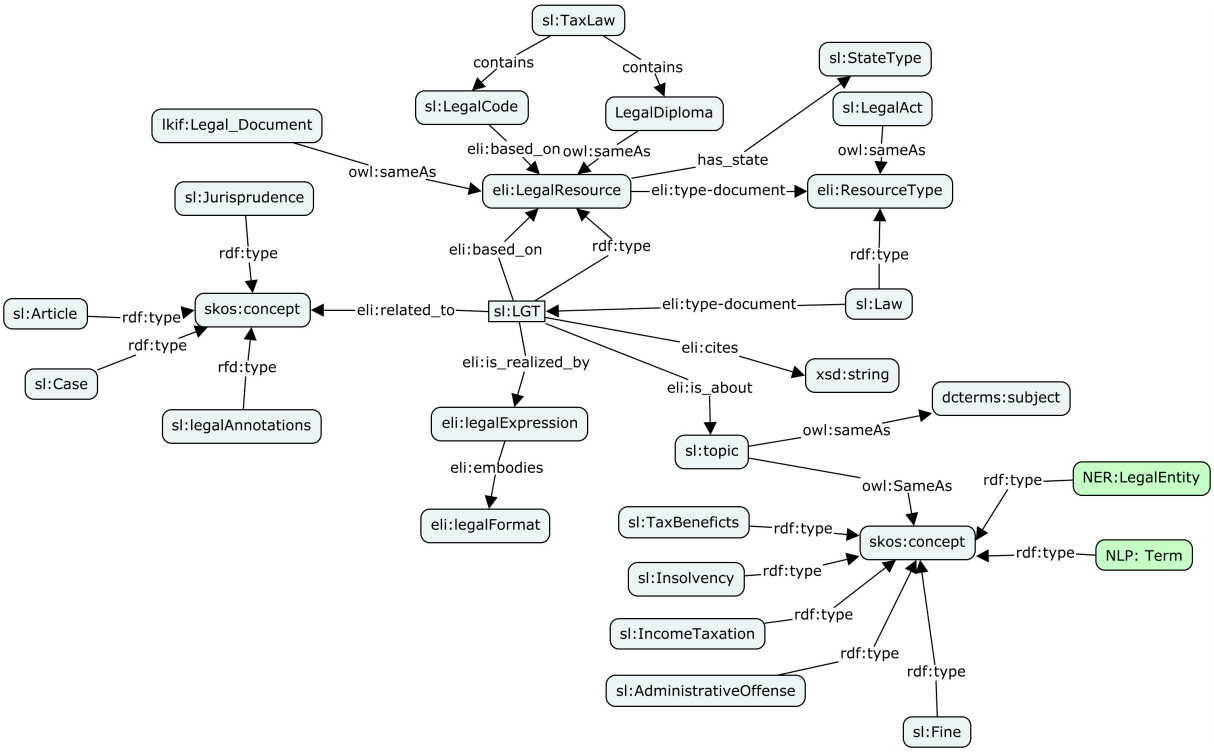


Figura 15: conceptualização final

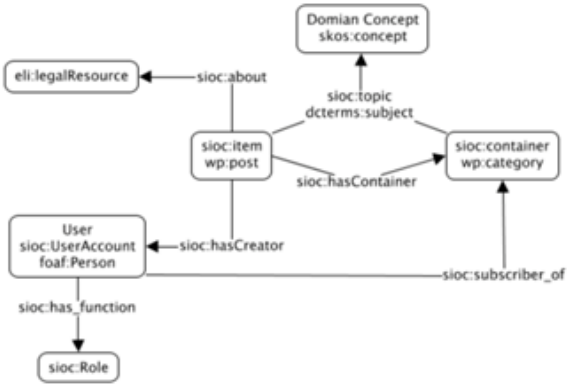


Figura 16: Modelo de Integração

De notar que os tópicos para classificação dos diplomas (através da relação *is-about* e/ou partes dos diploma legais, contêm uma lista base de conceitos definidos pelos especialistas. Adicionalmente, esses conceitos podem ser estendidos com o resultados dos algoritmos de NER e NLP.

4.2 *Pipeline* para extração terminológica

As técnicas de NLP, permitem processar grandes quantidades de texto não estruturado. Foi definida uma *pipeline* com o objetivo de extrair, do *corpus* dos documentos, um conjunto de palavras-chaves relevantes que resumem o conteúdo de um determinado documento. Esta tarefa não só pode identificar palavras relevantes no *corpus* do documento, como também pode ser utilizada para construir grupos de palavras (conhecidos nesta área por *ngrams* [58]) ou frases para identificar um contexto específico abordado num documento.

Na Figura 17, são apresentadas as várias sub-tarefas da *pipeline* desenvolvida para o processo de extração terminológica. A *pipeline* recebe como *input* o texto de um documento, este é posteriormente processado pela *pipeline* e o resultado final consiste numa lista de palavras, simples ou compostas, que resumem o conteúdo e são consideradas relevantes no texto do documento.

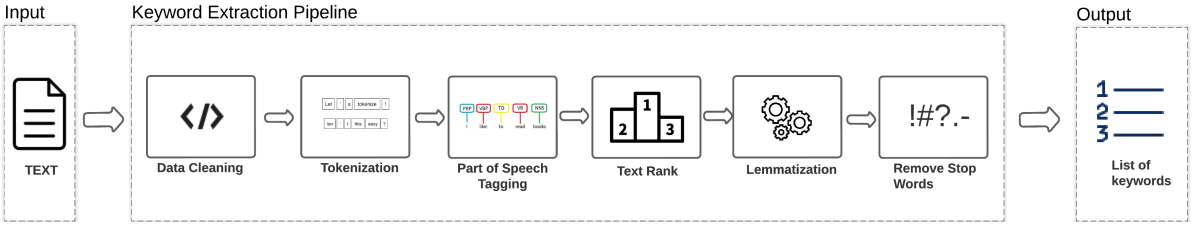


Figura 17: *Pipeline* para extração terminológica

Para implementar esta *pipeline* foi utilizada a biblioteca *spacy*[1], um biblioteca de NLP de alto desempenho que oferece diversos recursos. A *pipeline* começa por aplicar um processo de limpeza de dados. Como o *wordpress* armazena o texto com *tags HTML* é necessário inicialmente remover estas *tags*. Esta tarefa é importante no contexto de recuperação de informação, pois permite transformar o texto não-estruturado em um formato que as ferramentas de processamento de texto suportam [59].

De seguida, é aplicado o processo de *tokenization* e de *Part of Speeh Tagging*. Estes processos permitem analisar o texto ao nível granular. No primeiro, o texto é divido em unidades menores chamadas *tokens*. Um *token* pode ser uma palavra, um número ou pontuação. Enquanto, o segundo processo consiste em classificar cada *token* gramaticalmente de acordo com a sua função na frase. Estes *token* podem ser classificados como: verbo, adjetivo, advérbio, pronome, preposição, pontuação, entre outros.

[1] https://spacy.io/

Em seguida, é aplicado o *text rank*[2]. Este algoritmo, baseado em grafos para processamento de texto, permite extrair frases ou termos relevantes de um texto e atribui um valor de relevância ao respetivo termo no documento [60]. Os termos são apresentados do mais relevante para o menos relevante segundo o grau atribuído. Por fim, são aplicados processos de *lemmatization* para transformar os termos para a sua forma base (como por exemplo: tributações, transforma para o *lemma* tributação) e removidas *stop-words* da língua portuguesa para excluir possíveis termos não relevantes das palavras selecionadas.

A Figura 18 demonstra a utilização desta *pipeline* desenvolvida para extração terminológica. Como *input* recebe um texto (na figura à esquerda) e devolve como *output* um conjunto de termos (na figura à direita) e um valor de relevância que a biblioteca atribuí ao termo no texto. Na prática, o *output* desta *pipeline* permite identificar assuntos jurídicos no *corpus* dos documentos presentes na base de conhecimento, permitindo criar funcionalidades como: pesquisas por assuntos, criar sugestões em pesquisas livres, apresentar a opção de consultar recursos (diplomas, caso práticos, etc) com assuntos jurídicos similares.

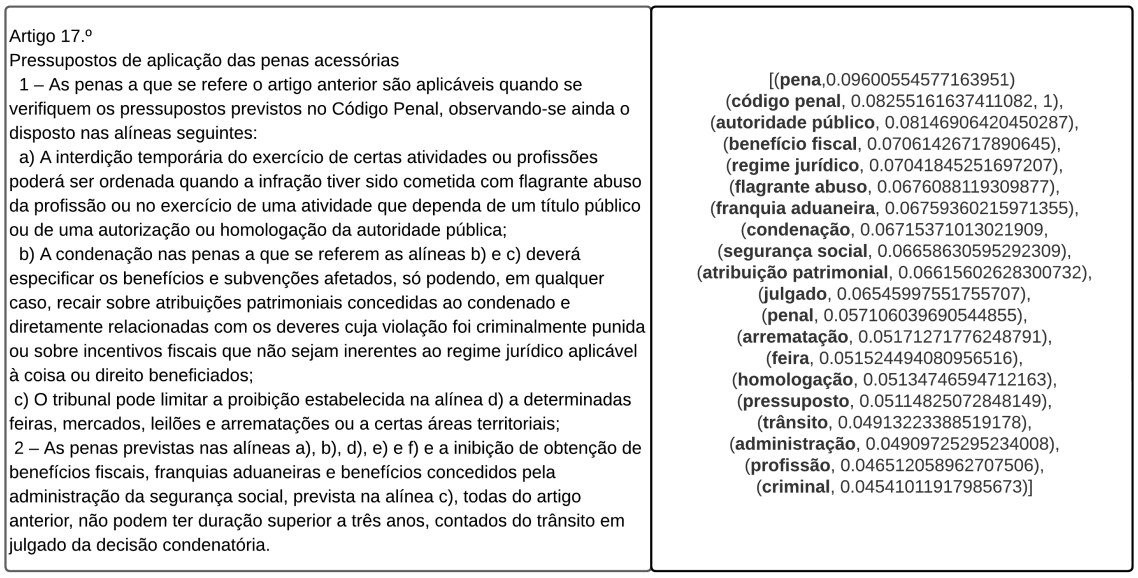


Figura 18: Extração terminológica

4.3 Modelo para identificação de entidades jurídicas

Para enriquecer as pesquisas realizadas à base de conhecimento, é importante que o sistema tenha a capacidade de extrair do *corpus*, não apenas uma lista de termos relevantes, mas entidades específicas do domínio. Para tal, é utilizada uma tarefa de NER, uma das principais

[2]https://pypi.org/project/pytextrank/

aplicações NLP responsável por identificar as entidades presentes no corpus dos documentos, classificando-as dentro de um conjunto de categorias pré-definidas para determinado contexto. Esta técnica tem como finalidade identificar entidades nos documentos jurídicos, adicionando-as como meta-informação aos documentos, possibilitando a criação de pesquisas facetadas, mais controladas em comparação com o processo de extração terminológica.

Foi disponibilizada uma ferramenta de anotação de texto, *doccano*[3], de forma a que os especialistas do domínio anotem entidades, com base num conjunto de *labels*. O processo de anotação do modelo NER consistiu em 3 iterações diferentes. Numa primeira iteração, foram anotados cerca de 100 documentos e definidas 74 *labels*. Estes documentos foram escolhidos aleatoriamente do *dataset* do OIF (incluindo códigos tributários, artigos de opinião, casos práticos, etc) e disponibilizados no *doccano*. As *labels* foram definidas tendo por base assuntos jurídicos e a ontologia do domínio. Contudo, no final do processo, as entidades foram anotadas de forma muito superficial, focando-se apenas num subconjunto das *labels* (essencialmente à base de referências para outros documentos).

Decidiu-se então realizar uma segunda iteração no processo de anotação, desta vez procurando identificar mais entidades e não apenas aquelas entidades mais óbvias à primeira vista. Contudo durante o processo, em conjunto com os especialistas do domínio e com base no seu *feedback*, chegou-se a conclusão que a quantidade de *labels* e o tamanho dos documentos (como os códigos tributários) tornava o processo exaustivo e dificultado.

Foi então decidido criar uma terceira iteração. Em reunião com os especialistas do domínio foram simplificadas as *labels* iniciais, passando de 74 para 18 *labels*. Procurou-se que as *labels* não fossem muito específicas e criar um *dataset* menos exaustivo de anotar. Na Tabela 2 são apresentadas as *labels* definidas para o processo de anotação, se estão associadas ao domínio fiscal ou ao domínio jurídico em geral e ainda alguns exemplos que os especialistas do domínio sugeriram.

Para resolver o problema verificado com o tamanho dos documentos no processo de anotação anterior, foram selecionados aleatoriamente diversos documentos do repositório do OIF e criado um *dataset* com 1050 documentos dos diferentes tipos (Atos legislativos, artigos de códigos tributários, artigos de opinião, anotações dos especialistas do domínio). Os documentos mais extensos foram excluídos ou foram divididos, por exemplo, os códigos tributários foram divididos por artigo e não foram anotadas algumas partes mais introdutórias que, segundo os

[3] https://doccano.github.io/doccano/

especialistas do domínio, não acrescentavam informação relevante.

A Figura 19(A), retrata o processo de anotação, a ferramenta de anotação permite criar um *dataset* de documentos com entidades especificas do domínio jurídico, para posteriormente, treinar um modelo de *machine learning*. Esta técnica tem como principal objetivo identificar entidades mais complexas que com abordagens mais simples, como *regex*, utilizadas muitas vezes para identificar entidades em tarefas de NER não seria possível, como organizações, pessoas, ou outras entidades presentes na Tabela 2. Mesmo referências a outros documentos com *regex* podem ser desafiantes, visto que, os documentos são escritos em linguagem natural (como casos práticos, artigos de opinião) e sem regras de estruturação rigorosas e com diferentes estilos entre os vários autores.

No final do processo de anotação, são exportadas as anotações num formato *JavaScript Object Notation Lines* (JSONL)[4]. Em seguida, foi necessário recorrer a um processo de transformação (através de um *script* em *python*[5]), das anotações produzidas de forma a que sejam utilizadas pelas bibliotecas de NLP. Neste contexto foi utilizado o *Prodigy*[6] para treinar o modelo NER. Este processo apenas é necessário porque, embora o *doccano* e o *Prodigy* utilizem o formato JSONL para armazenar as anotações, o *Prodigy* requer um estrutura específica para as anotações.

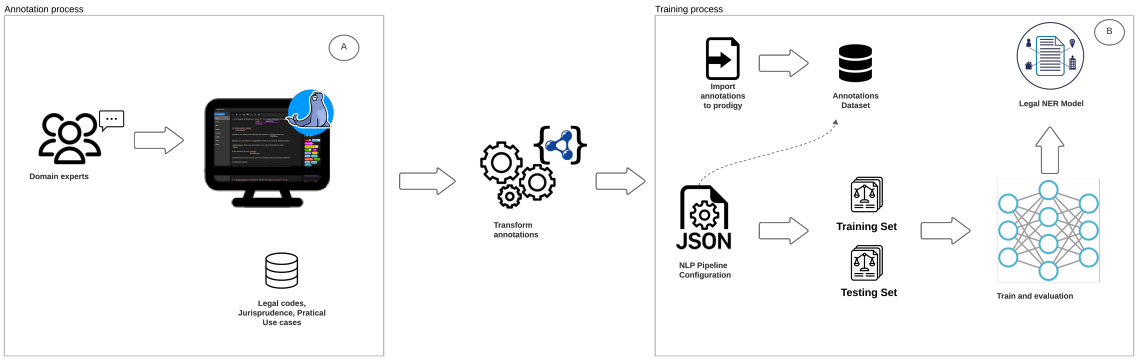


Figura 19: Treino do modelo *NER*

A Figura 19(B), apresenta o processo de treino do modelo NER com o *Prodigy*. As anotações depois de transformadas para o formato suportado pelo *Prodigy*, são armazenadas na sua base de dados como *datasets*. O treino do modelo com as anotações pode ser configurado num ficheiro *JSON*. Este ficheiro inclui as configurações do treino e os hiperparâmetros da

[4]Formato que armazena dados estruturados em linhas separadas, cada linha contém um objeto *JSON* independente

[5]https://www.python.org/

[6]https://prodi.gy/

Label	**Tipo de entidade**	**Exemplos**
Legislação	Geral	Lei n.º 2/98, Lei n.º 10/2022, Regime Geral das Infrações Tributárias, EBF, Estatuto dos benefícios fiscais, Decreto n.º 10/2022, Decreto n.º 10/2022
Artigo	Geral	Artigo 20º, Artigo 140º-B
Dívida	Geral	Dívida tributária, Dívida fiscal
Obrigação	Geral	Obrigação fiscal, Obrigação declarativa, Obrigação (contratos)
Decisão	Geral	Decisão do tribunal, Sentença, Decisão arbitrária, Decisão administrativa
Infração	Geral	Infração tributária, Infração
Proibição	Geral	Proibição do planeamento fiscal
Organização Jurídica	Geral	Tribunal, Tribunais administrativos e fiscais, Tribunais judiciais
Sujeito	Geral	Contribuinte, sujeito passivo, sujeito ativo
Procedimento	Fiscal	Procedimento tributário, Procedimento administrativo, Recurso hierárquico, Reclamação hierárquica, pedido de informação, informação vinculativa
Processo	Fiscal	Execução fiscal, Processo tributário, Processo administrativo, Ação declarativa, Ação executiva, Insolvência
Isenção	Fiscal	Isenção de imposto, Isenção de pagamento
Dedução	Fiscal	Deduções à coleta, Deduções específicas
Contencioso tributário	Fiscal	Multa, Execução fiscal, Ação judicial
Benefícios fiscais	Fiscal	Abatimento, Subsídio, Benefício fiscal para investimento
Administração do estado	Fiscal	Serviço de finanças, câmara municipal, Ministério dos negócios estrangeiros
Tributação	Fiscal	Tributação automóvel, Tributação património, Tributação de consumo, Tributação de rendimento
Garantias	Geral	Garantias contratuais, Garantias dos contribuintes, Garantias no processo

Tabela 2: Entidades definidas e respetivos exemplos

pipeline de NLP.

O *Prodigy* permite também usar uma configuração padrão para treinar o modelo de NER, tirando partido dos modelos pré-treinados no *spacy*, ou então através do ficheiro de configuração, é possível configurar: as componentes da pipeline de NLP, o modelo base para o treino do modelo, a percentagem de divisão do *dataset* para treino e teste ou o diretório para armazenar o modelo treinado.

O *dataset* construído para anotação de entidades jurídicas era constituído por 1050 documentos, contudo, apenas foram anotados 241 documentos, com aproximadamente 3800 anotações em 18 *labels*. Sendo o caso de estudo orientado ao domínio fiscal, neste contexto, em conjunto com os especialistas do domínio foram definidas *labels* para o domínio jurídico no geral e para o domínio fiscal. A Figura 20 apresenta a distribuição das *labels* no *dataset* anotado pelos especialistas do domínio. Algumas das *labels* (como sujeito e tributação) aparecem com mais frequência que as restantes no conjunto de documentos anotados.

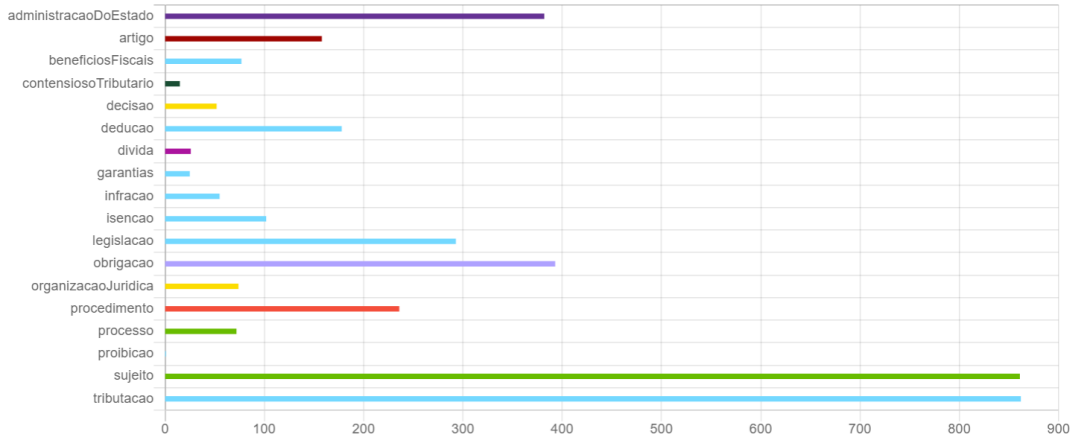


Figura 20: Distribuição das entidades anotadas

Foi configurada uma *pipeline* de NLP, tendo por base, o modelo pré-treinado do *spacy* para a língua portuguesa (pt_core_news_lg). Nesse ficheiro foi também configurado a divisão do *dataset* em 70/30 (70% dos dados para treino e 30% dos dados para teste), resultando em 169 documentos para treino e 72 documentos para teste. A evolução do treino é apresentado na Figura 21 através do gráfico, ao fim de 187 passagens pelo *dataset* (*epochs*) de treino o melhor resultado que foi possível atingir foi 53% (com 54.85% de precisão, 50.94% de *recall* e 52.82% de *f-measure*).

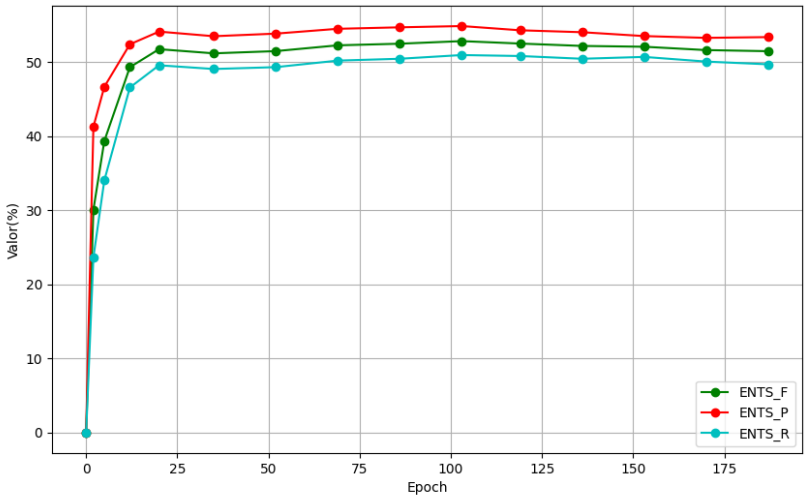


Figura 21: Gráfico de evolução no treino do modelo

4.4 Arquitetura da solução

A arquitetura da solução, apresentada na Figura 22, é resultado da revisão da literatura, do desenvolvimento dos protótipos e de sucessivas reuniões com os especialistas do domínio ao longo do projeto. Esta solução utiliza os artefactos anteriormente descritos e utiliza diversas ferramentas e tecnologias que permitem construir uma arquitetura facilmente escalável, independente e robusta.

Nesta arquitetura existem dois tipos diferentes de comunicações: comunicações assíncronas e comunicações síncronas. Para implementar as comunicações assíncronas foi utilizado o *Redis*[7], tirando partido do *Redis Streams*, uma estrutura de dados que foi projetada para lidar com o fluxo de mensagens e eventos em aplicações. Neste contexto existem três conceitos importantes:

- ***Stream*** - Um *stream* no *Redis* é uma sequência ordenada de eventos ou mensagens. Cada evento em uma *stream* é identificado por uma chave única, chamada de *Entry ID*. As mensagens são armazenadas na *stream* e consumidas pelos *consumers* de forma cronológica na *stream*. O exemplo de código 1 apresenta um exemplo de uma mensagem no *Redis Stream*;

```
1 {
2   "EntryID": "1692632086370-0",
```

[7] https://redis.io/

```
3    "message": "Hello, World!",
4    ...
5  }
```

Exemplos de Código 1: Exemplo de uma mensagem na stream

- ***Producer*** - O produtor é responsável por criar as mensagens. Cada mensagem é anexada ao final da *stream* e é-lhe atribuído um *ID* único. As mensagens têm o formato de *JSON*;
- ***Consumer*** - O consumidor é responsável por ler as mensagens da *stream*. Utiliza o último *ID* como apontador para a última mensagem lida na *stream*. Para além disso, o *Redis Stream* tem como principal vantagem permitir criar grupos de consumidores.

Nesta arquitetura, o serviço *orchestrator* funciona como uma espécie de *gateway* e permite a utilização dos serviços de conhecimento. Este serviço é responsável por controlar os *workflows* para processar os diferentes documentos, atua como um *producer* e publica mensagens em diferentes *streams* (identificadas por uma *key*). As mensagens são consumidas pelos *knowledge services* para realizar tarefas específicas. Por outro lado, os *knowledge services* também são *producers*, no final de realizar a respetiva tarefa publicam o resultado na *stream* para o *metadata broker* consumir.

Enquanto, *metadata broker* é responsável por gerir e enriquecer a base de conhecimento, armazenada no *neo4j*[8], com o *output* dos *knowledge services*, mapeando este *output* com a ontologia do domínio. Para além disto, contém toda a lógica de acesso à base de conhecimento para permite uma recuperação de informação eficiente e contextualizada.

Existem ainda dois serviços implementados em *python* e que fazem uso da biblioteca *FastAPI*[9] para implementar o padrão *REST*. Estes serviços utilizam mecanismos de comunicação síncrona, através de pedidos *Hypertext Transfer Protocol* (HTTP), para comunicar entre eles e para disponibilizar *endpoints* para outros componentes. Por exemplo, o *orchestrator* realiza um pedido HTTP *resquest* para criar um novo nó no grafo de conhecimento, o *metadata broker* devolve o resultado da operação, contudo, se a operação não for bem sucedida o serviço retorna o respetivo erro ao *content manager* e não executa o restante do *workflow*.

[8] https://neo4j.com/
[9] https://fastapi.tiangolo.com/

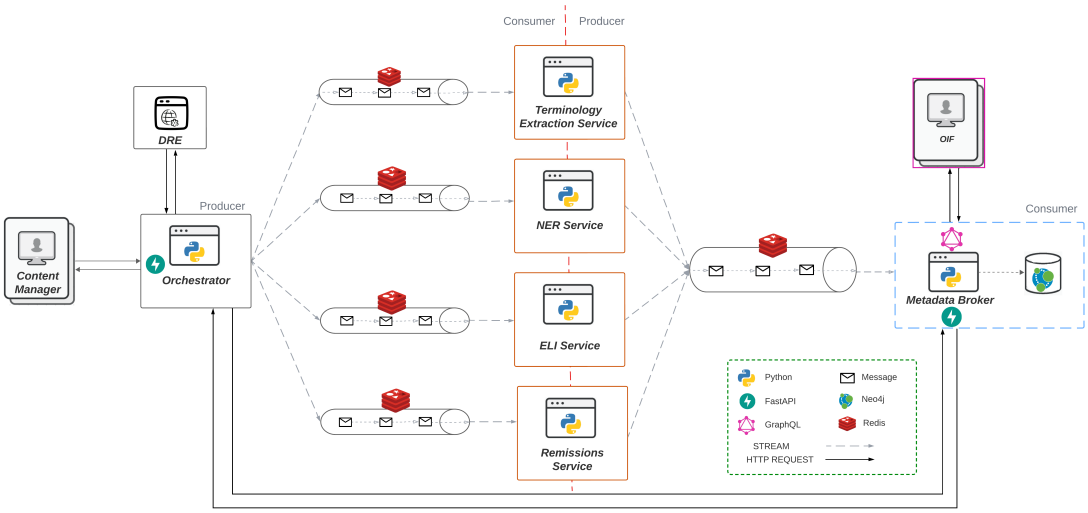


Figura 22: Arquitetura da solução

4.4.1 *Implementação dos Knowledge Services*

Os vários processos de recolha, tratamento e enriquecimento de informação, descritos anteriormente são encapsulados em serviços e mencionados como *knowledge services*. Fazem parte de uma arquitetura KaaS, composta por diversos serviços ou componentes distribuídos em várias camadas, cada um com um propósito específico, todos coordenados por um *orchestrator* encarregue de facilitar a interação com os diferentes serviços desenvolvidos e distribuir as tarefas por cada um dos serviços.

Os *knowledge services*, na Figura 22 representados a laranja, foram implementados em *python*. Dependendo do propósito estes serviços utilizam diferentes bibliotecas, os serviços desenvolvidos foram:

- ***Orchestrator*** - Este serviço não é considerado um *knowledge service*, contudo, possuí um conjunto de *endpoints* que permitem processar os documentos jurídicos para alimentar a base de conhecimento. É neste componente que se encontram implementados os *workflows*[10] para processar cada tipo de documento jurídico na base de conhecimento. Este serviço comunica com o *metadata broker* através de comunicações síncronas, com pedidos HTTP, para criar os nós no grafo de conhecimento e com os *knowledge service* por comunicações assíncronas (*streams*) para processar o texto dos documentos no respetivo serviço. Para além disso, é responsável por extrair informações de outros vocabulários, como os meta-dados dos atos legislativos, com um técnica de *web scraping*[11],

[10] Sequências de tarefas organizadas de maneira lógica e automatizada para alcançar um objetivo específico
[11] Processo de extrair automaticamente informações de sites da web

do DRE, evitando que sejam feitos múltiplos pedidos quando vários serviços precisam de dados da mesma fonte;

- ***Terminology Extraction Service*** - Este serviço tem implementada a *pipeline* para extração terminológica apresentada na secção 4.2. Este serviço mantém-se à ”escuta” de mensagens publicadas na *stream* com o *orchestrator*, cada mensagem contém o identificador do documento e texto do documento para ser processada na *pipeline* de NLP e extrair os termos relevantes. No final, publica o resultado deste processo na *stream* para o *metadata broker* consumir;

- **NER *Service*** - Este serviço tem como objetivo identificar entidades no texto dos documentos que recebe por *stream*, faz uso do modelo NER treinado e explicado na secção 4.3. Usa o *spacy* para importar o modelo treinado, processar o texto e no final publica na *stream* do *metadata broker* as entidades identificadas;

- **ELI *Service*** - Este serviço recebe por *stream* os meta-dados do ELI nos atos legislativos e o identificador do documento. Os meta-dados são processados numa função e publicados na *stream* do *metadata broker* para enriquecer o grafo de conhecimento;

- ***Remissions Service*** - Este serviço tem como objetivo extrair remissões padrão para outros documentos jurídicos no texto do documento. Tal como os outros serviços recebe por *stream* o identificador e o texto do documento e através de expressões regulares extrai referências para outros tipos de documento. No final, publica na *stream* do *metadata broker* para criar relacionamentos entre os documentos. Alguns exemplos remissões: Lei n.º 56/2023, Portaria n.º 265/2023, Art.º 92 da LGT, etc.

4.4.2 Integração do *graphQL*

Por fim, para utilizar o grafo de conhecimento e tirar partido do resultado dos *knowledge services*, foi implementado o *graphQL*[12] no componente *metadata broker*. Esta linguagem de consulta, em vez de receber os dados predefinidos, como no padrão *REST*, permite aos clientes (neste contexto o componente do OIF) especificar na *query* exatamente quais atributos do modelo de dados desejam recuperar. Isto evita problemas de super ou sub-seleção de dados que ocorrem em serviços *REST* tradicionais.

[12] https://graphql.org/

Além disso, o modelo de dados definido funciona como uma hierarquia. Permite que o utilizador especifique o tipo de documentos que deseja recuperar da base de conhecimento, e consequentemente, desses documentos quais os atributos que deseja receber. O utilizador define a *query* através da linguagem *graphQL*, e a biblioteca internamente faz o mapeamento para as *queries* desenvolvidas *cypher*.

O exemplo de código 2 apresenta um exemplo de uma *query* com o *graphQL*. Neste exemplo, apenas são retornados documentos da legislação, mais em concreto, decretos-leis e leis e jurisprudências com os respetivos atributos definidos para cada um. A principal vantagem do uso deste tipo de abordagens é oferecer a capacidade de personalizar os resultados obtidos consoante as necessidades de utilização.

```
query {
  legislacao() {
    decretoLei {
      documentId
      title
    }
    leis {
      documentId
    }
  }
  jurisprudencia() {
    documentId
  }
}
```

Exemplos de Código 2: Exemplo de uma query em graphQL

Capítulo 5

Caso de Estudo

Os atuais mecanismos de recuperação demonstram-se ineficientes e pouco úteis para a prática das atividades jurídicas no dia-a-dia dos utilizadores da plataforma do *Lexit*. Os utilizadores não conseguem obter determinada informação mediante determinadas pesquisas. Para além disso, não conseguem definir ou filtrar o tipo de recurso textual que pretendem. Por isso, na maioria das vezes, recorrem a um *chat* que a plataforma dispõe. Contudo, este *chat* só funciona se estiver alguém do lado do *Lexit online* e a responder às questões que são colocadas nesse *chat*. Caso contrário, os utilizadores têm de aguardar por uma resposta que poderá não existir.

Algumas das perguntas frequentes que os utilizadores fazem através do *chat* são:

1. Em que situações recebo o reembolso do IVA?
2. É possível pagar uma dívida tributária em prestações?

Este capítulo incide fundamentalmente na experimentação da abordagem KaaS descrita nos capítulos anteriores. Pretende-se validar a resposta da solução face as questões apresentadas anteriormente que tipicamente o *Lexit* não responde. Por isso, na secção 5.1 é feita uma caracterização do *corpus* e na secção 5.2 é apresentado os resultados obtidos através da solução desenvolvida.

5.1 Caracterização do *corpus*

A atual *stack* tecnológica da plataforma *Lexit* consiste numa arquitetura típica de uma aplicação *web* em *wordpress*[1]. O *wordpress* é considerado um *Content Management System* (CMS) e tem como principal vantagem permitir que um utilizador sem qualquer conhecimento sobre desenvolvimento de aplicações *web* consiga facilmente criar e atualizar o conteúdo do *website* [61].

Todo o conteúdo do *Lexit*, incluindo configurações do próprio *wordpress*, está armazenado em duas tabelas relacionais (*wp_posts* e *wp_postmeta*). Para filtrar e selecionar tabelas e colunas relevantes, foi inicialmente aplicado um pré-processamento. Após este processo de seleção, foram identificados 13229 documentos com um tamanho total de 36,93 *megabytes*. Em específico os tipos de documentos são:

- **artigo** (1430 documentos) - Artigos de opinião criados pelos especialistas do domínio sobre diversos assuntos, têm como principal objetivo auxiliar na compressão de determinados assuntos jurídicos;
- **caso-pratico** (193 documentos) - Documentos usados como exemplo para ilustrar a aplicação das leis e princípios do direto em situações reais;
- **diario-da-republica** (7172 documentos) - Diplomas publicados, nos dias úteis, no DRE que revogam ou definem novas leis em vigor;
- **diploma** (3492 documentos) - Atos jurídicos publicados nos diplomas do DRE, que através de leis e decretos-leis alteram a legislação em vigor;
- **doutrina** (755 documentos) - Opiniões, pensamentos, teorias que são ensinados e seguidos por professores, alunos e profissionais da área do direito que fundamentam decisões ou posições que são tomadas relativas à administração ou ao direito administrativo;
- **euribor**(109 documentos) - Notícias sobre a variação da taxa da *euribor*;
- **jurisprudência** (78 documentos) - Decisões sobre interpretações das leis feitas pelos tribunais.

[1] https://pt.wordpress.org/

Para além deste conjunto de documentos, existe ainda a estrutura dos códigos legais e as respetivas anotações de cada artigo(s) dos códigos tributários. A estrutura dos códigos legais contém 18989 instâncias em que aproximadamente 14674 são artigos/linhas de artigos e o restante são instâncias que estruturam os códigos (como títulos, secções ou subsecções). Existem ainda 3690 instâncias de anotações do(s) especialista(s) do domínio para os artigos dos códigos tributários.

O *corpus* do documentos é constituído por, aproximadamente, 5255260 *tokens*/palavras presentes em 28000 documentos do domínio fiscal português. No *corpus* dos documentos do domínio fiscal destacam-se os seguintes elementos:

- **Referências para legislação**, isto é, referências para diplomas publicados no DRE, como leis, decreto-lei. Como por exemplo: Decreto-Lei n.º 72/2022; Lei n.º 12/2022;
- **Referências para códigos tributários** que complementam ou justificam determinada afirmação. Como por exemplo: Lei Geral Tributária, Código do Imposto Selo ;
- **Referências a artigos** específicos dentro do próprio documento ou/e para artigos de outros documentos, como legislação ou códigos tributários. Como por exemplo: artigo 12.º do Regime Geral das Infrações Tributárias; artigo 4.º da Lei n.º 12/2022;
- **Agentes**, isto inclui sujeitos ou organizações públicas ou privadas. Como por exemplo: Sujeito passivo, Segurança Social;
- **Referências a jurisprudências** para justificar decisões ou dar exemplos práticos da aplicação da lei. Por exemplo: Acórdão n.º 3/2021 do Supremo Tribunal Administrativo;
- **Valores monetários** incluem diferentes valores, como por exemplo, de indemni-zações ou multas.
- **Datas e períodos de tempo** definindo datas ou períodos de tempo específicos, como para o pagamento de coimas ou entrega de documentos.

Na Figura 23, é apresentado um excerto de texto presente num dos documentos com alguns dos elementos apresentados. A maioria dos documentos jurídicos apresenta um *corpus* complexo e com propriedades muito específicas do domínio. Por essa razão, efetuar uma pesquisa por um termo, pode devolver centenas de documentos que podem ou não estar de acordo com o objetivo e o contexto da pesquisa.

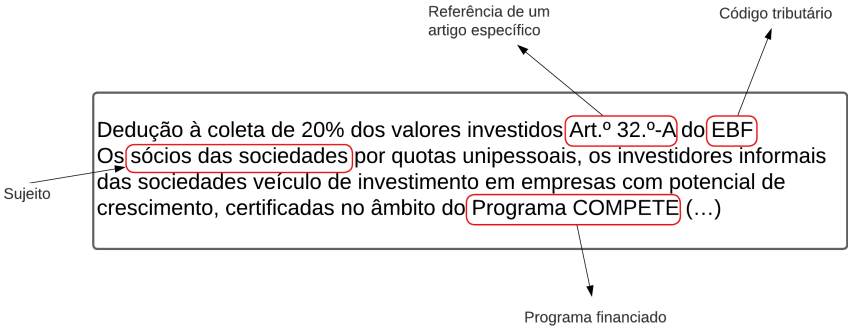


Figura 23: Excerto de um documento fiscal

5.2 Exploração e recuperação de informação

O *Lexit* disponibiliza na sua plataforma uma ferramenta de pesquisa baseada em texto. Permite que qualquer utilizador da plataforma introduzir um ou vários termos (ainda que limitados em número, por exemplo, a introdução de frases ou perguntas não é suportada) e obter como resultado um conjunto de documentos. Com base no *feedback* dos especialistas do domínio, os resultados obtidos atualmente através desta ferramenta são excessivos e praticamente aleatórios. Os resultados são apresentados sem uma estrutura adequada (como por exemplo uma hierarquia), são apresentados por ordem aleatória e não é possível filtrar por um tipo de recurso textual em específico (como legislação, artigos de opinião, casos práticos).

Para avaliar os resultados dos serviços KaaS foi processado o *corpus* apresentado na secção anterior, cada tipo de recurso textual é processado com base no *workflow* apresentado no anexo A. O principal objeto é avaliar a capacidade da solução na resposta a questões jurídicas através da exploração do grafo de conhecimento, seja uma questão mais complexa ou uma simples pesquisa por assunto jurídico, para além disso, também foi desenvolvida uma *query* que permite obter documentos relacionados a um determinado documento. O principal objetivo destas *queries* é auxiliar os utilizadores do *Lexit* no processo de recuperação de informação contextualizada, mitigando os problemas atuais no processo de recuperação de informação descritos anteriormente.

A tabela 3, apresenta um exemplo de uma pesquisa por assunto/termo do domínio jurídico. Neste caso, o sistema recebe como *input* uma *query* com o termo ”IRS” e o tipo de recurso textual desejado (artigo de opinião e legislação) e devolve como *output* 22 documentos relacionados com esse assunto/termo. Estes documentos são obtidos com base nos relacionamentos presente na base de conhecimento, com por exemplo o *is-about*.

	Input	*Query*	*Output*

Pesquisa por assunto	”IRS”	query MyQuery {
		artigoOpiniaoByName
		(name: ”IRS”) {
		documentId
		}
		legislacaoByName
		(name: ”IRS”) {
		artigos {
		documentId
		}
		}
		}
			(22 documentos)
			{
			”data”: {
			”artigoOpiniaoByName”:
			[
			{
			”documentId”: 100392
			},
			(...)
			],
			”legislacaoByName”: {
			”artigos”: [
			{
			”documentId”: 155439
			},
			(...)
			]
			}
			}
			}

Tabela 3: Pesquisa por assunto

Além dos problemas mencionados, ao consultar um determinado documento no *Lexit* não são apresentadas sugestões de outros documentos que podem ser úteis ou que complementam o documento em questão. A tabela 4, apresenta um exemplo de pesquisa que pode ser utilizada para obter documentos relacionados, o sistema recebe como *input* uma *query* onde é definido o identificador do documento (*documentId*) e devolve como *output* um conjunto de documentos (neste caso, apenas decretos-lei) que se relacionem com esse documento. Os documentos podem ser obtidos com base em vários relacionamentos como: *cite* (documentos mencionados no *corpus* deste documento), *cited-by* (documentos que citam este documento) ou *is-about* (partilham assuntos em comum).

	Input	*Query*	*Output*

Documentos relacionados	9888	query MyQuery {
		legislacao
		(documentId: 9888) {
		decretoLei {
		documentId
		}
		}
		}
			(6 documentos)
			{
			”data”: {
			”legislacao”: {
			”decretoLei”: [
			{
			”documentId”: 90014
			},
			{
			”documentId”: 90018
			},
			(...)
			]
			}
			}
			}

Tabela 4: Pesquisa por documentos relacionados

O *Lexit* como referido anteriormente apresenta um limite no número de termos que pode ser utilizado pelo mecanismo de pesquisa de informação, se o utilizador introduzir uma frase mais complexa acaba por não obter resultado, por isso, recorrem frequentemente ao *chat* da plataforma. Foi testada a capacidade da solução desenvolvida para responder as *queries* apresentadas no início deste capítulo. No *metadata broker* (componente que implementa a linguagem *graphql* para acesso à base conhecimento) foi implementada uma *query* que inicialmente processa o *input* do utilizador e aplica o processo de extração terminológica para identificar os termos mais importantes dessa *query*.

A tabela 5 apresenta o resultado depois de executadas estas *queries* na solução desenvolvida. Na tabela foram usados o número 1 (Em que situações recebo o reembolso do IVA?) e 2 (É possível pagar uma dívida tributária em prestações?) para representar mais facilmente o *input* do utilizador na *query*.

	Input	*Query*	*Output*

Pesquisa	1	`query MyQuery {`
		`  legislacao`
		`  (prompt: "1") {`
		`  artigos {`
		`  documentId`
		`  }`
		`  }`
		`}`
			`(4 documentos)`
			`{`
			`  "data": {`
			`  "legislacao": {`
			`  "artigos": [`
			`  {`
			`  "documentId": 153506`
			`  },`
			`  (...)]`
			`  }`
			`  }`
			`}`

Pesquisa	2	`query MyQuery {`
		`  legislacao`
		`  (prompt: "2") {`
		`  artigos {`
		`  documentId`
		`  }`
		`  }`
		`}`
			`(2 documentos)`
			`{`
			`  "data": {`
			`  "legislacao": {`
			`  "artigos": [`
			`  {`
			`  "documentId": 155389`
			`  },`
			`  {`
			`  "documentId": 155378`
			`  }]`
			`  }`
			`  }`
			`}`

Tabela 5: Pesquisa com base em perguntas

Em suma, a solução desenvolvida suprime alguns dos problemas atuais do *Lexit*, este mecanismo permite especificar o tipo de recurso textual desejado num cenário de pesquisa, consultar documentos relacionados, a implementação de pesquisas facetadas e ainda a recuperação de informação com *queries* mais completas.

Capítulo 6

Conclusões e Trabalho Futuro

6.1 Reflexão Crítica

A recuperação de informação em domínios altamente complexos, como o jurídico, apresenta uma série de desafios não só na compreensão das especificidades do domínio mas também na interpretação e extração de conhecimento do seu *corpus*, extremamente específicos e habitualmente ambíguos. Esta problemática é inerente à natureza multifacetada do campo jurídico e exige abordagens inovadoras e tecnológicas para ser enfrentada com sucesso.

O domínio jurídico é caracterizado por uma linguagem densa e técnica, repleta de termos e conceitos que não são facilmente compreendidos por leigos[1]. Essa complexidade linguística é agravada pela evolução constante do direito, que resulta numa constante criação e revisão de leis, regulamentos e jurisprudência. A ambiguidade é outro desafio comum, já que muitas vezes um único termo ou frase pode ter diferentes interpretações em contextos jurídicos distintos.

Este documento reflete vários destes aspetos. O conteúdo utilizado como *input* do trabalho abrange uma série de documentos que incluem jurisprudência, artigos de opinião ou sínteses anotadas. Todos estes documentos possuem um léxico complexo associado a contextos muito específicos e com diversas referências a outros recursos legais, como códigos ou diplomas. Como consequência, as pesquisas realizadas no âmbito do *Lexit* sofrem de muitas limitações na resposta às pesquisas dos utilizadores, devolvendo em alguns casos demasiada

[1] Pessoas sem os conhecimentos necessários

informação e principalmente informação descontextualizada ou mesmo não relacionada.

Foi construída uma solução tendo por base princípios semânticos e princípios de IA. Para isso foi desenvolvida uma ontologia do domínio fiscal, envolvendo os especialista de domínio de forma a contextualizar os conceitos do domínio e enquadrando-os com os conceitos da legislação existente (aproveitando vocabulários já existentes e utilizados neste contexto, como é o caso do ELI). As técnicas de NLP são utilizadas para analisar e compreender documentos, identificar padrões de linguagem jurídica e categorizar informações relevantes, principalmente relacionadas com entidades no domínio jurídico português. Uma técnicas de NER foi explorada para treinar modelos capazes de reconhecer e extrair informações específicas a partir do input dos especialistas que colaboraram com o desenvolvimento deste projeto através do seu conhecimento e visão específica sobre os temas. Este processo visava uma melhor integração do sistema e a personalização dos resultados de acordo com a perspetiva dos especialistas do OIF.

Os diversos componentes do projeto resultam num conjunto de serviços seguindo uma abordagem KaaS que entrega ao utilizador dados contextualizados. Cada serviço tem o seu propósito, comunicando entre si para garantir a recuperação de informação eficiente. Além disso, do ponto de vista de interoperabilidade, estes serviços permitem uma integração não intrusiva com a plataforma do OIF (recorrendo a linguagem *graphql*), ao mesmo tempo que disponibiliza os meios necessários para garantir a escalabilidade dos serviços e a potencial integração de novos serviços sem comprometer a infraestrutura existente.

De uma forma geral, os objetivos definidos inicialmente foram atingidos. A solução desenvolvida, em comparação com o atual mecanismo de recuperação de informação do *Lexit*, facilita o processo de recuperação de informação. Existe uma estrutura de conhecimento que garante a organização dos dados e suporta o processo de recuperação de informação, permitindo a definição dos recursos textuais a consultar, a implementação pesquisas facetadas e a utilização de *queries* mais complexas

A utilização de técnicas de NER para processar o *input* fornecido pelos especialistas do domínio e personalizar os resultados de acordo com a sua perspetiva foi o componente que revelou mais desafios. Apesar da existência de uma ferramenta *web* acessível que os especialistas poderiam utilizar para anotar o *corpus* processado, os resultados não atingiram a expetativa inicialmente definida. Apesar dos esforços, foi difícil recolher um número significativo de anotações e com a coerência que inicialmente era prevista, tendo naturalmente em

consideração o ponto de vista do processo de recuperação de informação. Como resultado, não foi possível compreender todo o potencial que a abordagem suportada por técnicas de NER tendo por base o *input* dos utilizadores poderia disponibilizar.

6.2 Trabalho Futuro

Como trabalho futuro é pretendido melhorar os resultados do modelo de NER, aumentar o número de anotações e analisar a evolução do treino do modelo. Pretende-se também tirar partido da ontologia que suporta o mecanismo de recuperação de informação e utilizar os mecanismos de *reasoning* para aumentar a qualidade no processo de recuperação de informação.

Além disso, é pretendido integrar a solução desenvolvido com a plataforma do *Lexit*. Contudo, é importante garantir os dados são corretamente processados na solução, para isso o ideal será futuramente, e antes desta integração, desenvolver uma *user interface* para gerir os resultados dos KaaS e a base de conhecimento, para facilitar a interação dos responsáveis por gerir o conteúdo do OIF com a solução desenvolvida e apresentada nesta dissertação.

Bibliografia

[1] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *Mobile Networks and Applications*, vol. 19, pp. 171–209, 2014.

[2] R. Sherman, *Chapter 1. The Business Demand for Data, Information, and Analytics*, pp. 3–19. 12 2015.

[3] G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, and R. Rosati, *Using Ontologies for Semantic Data Integration*, pp. 187–202. Cham: Springer International Publishing, 2018.

[4] M. Gomes, B. Oliveira, and C. Sousa, "Enriching legal knowledge through intelligent information retrieval techniques: A review," in *Progress in Artificial Intelligence* (G. Marreiros, B. Martins, A. Paiva, B. Ribeiro, and A. Sardinha, eds.), (Cham), pp. 119–130, Springer International Publishing, 2022.

[5] G. Sanchez, "Sentence boundary detection in legal text," in *Proceedings of the natural legal language processing workshop 2019*, pp. 31–38, 2019.

[6] M. Bajcic, "Conceptualization of legal terms in different fields of law: The need for a transparent terminological approach," *Research in language*, vol. 9, 2011.

[7] M. Frické, "Big data and its epistemology," *Journal of the association for information science and technology*, vol. 66, no. 4, pp. 651–661, 2015.

[8] C. Devins, T. Felin, S. Kauffman, and R. Koppl, "The law and big data," *Cornell JL & Public Policy*, vol. 27, p. 357, 2017.

[9] H. Schütze, C. D. Manning, and P. Raghavan, *Introduction to information retrieval*, vol. 39. Cambridge University Press Cambridge, 2008.

[10] P. Johannesson and E. Perjons, *An introduction to design science*, vol. 10. Springer, 2014.

[11] A. Hevner and S. Chatterjee, *Design Science Research in Information Systems*, pp. 9–22. Boston, MA: Springer US, 2010.

[12] J. Venable and R. Baskerville, "Eating our own cooking: Toward a more rigorous design science of research methods," *Electronic Journal of Business Research Methods*, vol. 10, no. 2, pp. pp141–153, 2012.

[13] M. Shaker, H. Ibrahim, A. Mustapha, and L. N. Abdullah, "A framework for extracting information from semi-structured web data sources," in *2008 Third International Conference on Convergence and Hybrid Information Technology*, vol. 1, pp. 27–31, 2008.

[14] S. T. de Souza, "Modelagem de domínios em sistemas de organização do conhecimento (soc): uma investigação em tesauros e ontologias para a informação legislativa," 2017.

[15] R. Rocha Souza, D. Tudhope, and M. Almeida, "Towards a taxonomy of kos: Dimensions for classifying knowledge organization systems," *KNOWLEDGE ORGANIZATION*, vol. 39, pp. 179–192, 01 2012.

[16] T. Saracevic, "Information science," *Journal of the American Society for Information Science*, vol. 50, no. 12, pp. 1051–1063, 1999.

[17] G. Kuck, "Tim berners-lee's semantic web," *South African Journal of Information Management*, vol. 6, 12 2004.

[18] J. Euzenat and P. Shvaiko, *Ontology matching: Second edition*. 10 2013.

[19] D. Adams, S. Milton, E. Kazmierczak, and J. Lindenthal, "Thesaurus and ontology structure: Formal and pragmatic differences and similarities: Thesaurus and ontology structure: Formal and pragmatic differences and similarities," *Journal of the Association for Information Science and Technology*, vol. 66, 12 2014.

[20] A. Pellini and H. Jones, "Knowledge taxonomies: a literature review–research reports and studies," 2011.

[21] P. Lambe, "Organising knowledge: Taxonomies, knowledge and organisational effectiveness," *Organising Knowledge: Taxonomies, Knowledge and Organisational Effectiveness*, pp. 1–277, 02 2007.

[22] J. Barrasa, A. E. Hodler, and J. Webber, "Knowledge graphs data in context for responsive businesses," 2021.

[23] R. Studer, V. R. Benjamins, and D. Fensel, “Knowledge engineering: Principles and methods,” *Data & Knowledge Engineering*, vol. 25, pp. 161–197, 3 1998.

[24] V. Nguyen, “Ontologies and information systems: a literature survey,” 2011.

[25] N. Noy and D. Mcguinness, “Ontology development 101: A guide to creating your first ontology,” *Knowledge Systems Laboratory*, vol. 32, 01 2001.

[26] M. Almeida and M. Bax, “Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção,” *Ciência da Informação*, vol. 32, 12 2003.

[27] V. Maniraj and S. Ramakrishnan, “Ontology languages – a review,” *International Journal of Computer Theory and Engineering*, vol. 2, pp. 887–891, 01 2010.

[28] K. Munir and M. Sheraz Anjum, “The use of ontologies for effective knowledge modelling and information retrieval,” *Applied Computing and Informatics*, vol. 14, no. 2, pp. 116–126, 2018.

[29] P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, S. Rudolph, *et al.*, “Owl 2 web ontology language primer,” *W3C recommendation*, vol. 27, no. 1, p. 123, 2009.

[30] I. Robinson, J. Webber, and E. Eifrem, “Graph databases - new opportunities for connected data,” *Joe Celko’s Complete Guide to NoSQL*, pp. 27–46, 2015.

[31] C. Wang, H. Yu, and F. Wan, “Information retrieval technology based on knowledge graph,” in *2018 3rd International Conference on Advances in Materials, Mechatronics and Civil Engineering (ICAMMCE 2018)*, pp. 291–296, Atlantis Press, 2018.

[32] S. Purohit, N. Van, and G. Chin, “Semantic property graph for scalable knowledge graph analytics,” in *2021 IEEE International Conference on Big Data (Big Data)*, pp. 2672–2677, 2021.

[33] R. Angles, H. Thakkar, and D. Tomaszuk, “Mapping rdf databases to property graph databases,” *IEEE Access*, vol. 8, pp. 86091–86110, 2020.

[34] M. Saad, Y. Zhang, J. Tian, and J. Jia, “A graph database for life cycle inventory using neo4j,” *Journal of Cleaner Production*, vol. 393, p. 136344, 2023.

[35] R. Hoekstra, J. Breuker, M. Di Bello, A. Boer, *et al.*, “The lkif core ontology of basic legal concepts.,” *LOAIT*, vol. 321, pp. 43–63, 2007.

[36] T. Francart, J. Dann, R. Pappalardo, C. Malagon, and M. Pellegrino, “The european legislation identifier,” *Knowledge of the Law in the Big Data Age*, vol. 317, pp. 137–148, 2019.

[37] D. E. O’Leary, “Artificial intelligence and big data,” *IEEE intelligent systems*, vol. 28, no. 2, pp. 96–99, 2013.

[38] A. McCallum, K. Nigam, J. Rennie, and K. Seymore, “Building domain-specific search engines with machine learning techniques,” in *Proceedings of AAAI ’99 Spring Symposium on Intelligent Agents in Cyberspace*, March 1999.

[39] R. Dale, H. Moisl, and H. Somers, *Handbook of natural language processing*. CRC press, 2000.

[40] S. Sanyal, S. Hazra, S. Adhikary, and N. Ghosh, “Resume parser with natural language processing,” *International Journal of Engineering Science*, vol. 4484, 2017.

[41] P.-H. Chen, “Essential elements of natural language processing: what the radiologist should know,” *Academic radiology*, vol. 27, no. 1, pp. 6–12, 2020.

[42] N. S. Khan, A. Abid, and K. Abid, “A novel natural language processing (nlp)–based machine translation model for english to pakistan sign language translation,” *Cognitive Computation*, vol. 12, pp. 748–765, 2020.

[43] M. R. Hasan, M. Maliha, and M. Arifuzzaman, “Sentiment analysis with nlp on twitter data,” in *2019 international conference on computer, communication, chemical, materials and electronic engineering (IC4ME2)*, pp. 1–4, IEEE, 2019.

[44] R. Oshikawa, J. Qian, and W. Y. Wang, “A survey on natural language processing for fake news detection,” *arXiv preprint arXiv:1811.00770*, 2018.

[45] S. Singh, “Natural language processing for information extraction,” *arXiv preprint arXiv:1807.02383*, 2018.

[46] D. Nadeau and S. Sekine, “A survey of named entity recognition and classification,” *Lingvisticae Investigationes*, vol. 30, 08 2007.

[47] M. R. Dias, “Discovery of sensitive data with natural language processing,” Master’s thesis, 2019.

[48] L. Li, “The effects of trust and shared vision on inward knowledge transfer in subsidiaries’ intra-and inter-organizational relationships,” *International Business Review*, vol. 14, no. 1, pp. 77–95, 2005.

[49] R. Giri, Y. Porwal, V. Shukla, P. Chadha, and R. Kaushal, “Approaches for information retrieval in legal documents,” in *2017 Tenth International Conference on Contemporary Computing (IC3)*, pp. 1–6, IEEE, 2017.

[50] M. Ceci and A. Gangemi, “An owl ontology library representing judicial interpretations,” *Semantic Web*, vol. 7, pp. 229–253, 03 2016.

[51] S. Gostojić, B. Milosavljević, and Z. Konjović, “Ontological model of legal norms for creating and using legislation,” *Computer Science and Information Systems*, vol. 10, no. 1, pp. 151–171, 2013.

[52] A. Gangemi, M.-T. Sagri, and D. Tiscornia, *A Constructive Framework for Legal Ontologies*, pp. 97–124. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005.

[53] G. Carnaz, V. B. Nogueira, M. Antunes, and N. F. Ferreira, “Named-entity recognition for portuguese police reports,”

[54] R. Giri, Y. Porwal, V. Shukla, P. Chadha, and R. Kaushal, “Approaches for information retrieval in legal documents,” in *2017 Tenth International Conference on Contemporary Computing (IC3)*, pp. 1–6, 2017.

[55] B. Fawei, J. Z. Pan, M. Kollingbaum, and A. Z. Wyner, “A semi-automated ontology construction for legal question answering,” *New Generation Computing*, vol. 37, pp. 453–478, 2019.

[56] F. O. Omotayo, “Knowledge management as an important tool in organisational management: A review of literature,” *Library Philosophy and Practice*, vol. 1, no. 2015, pp. 1–23, 2015.

[57] S. Xu and W. Zhang, “Knowledge as a service and knowledge breaching,” in *2005 IEEE International Conference on Services Computing (SCC’05) Vol-1*, vol. 1, pp. 87–94, IEEE, 2005.

[58] M. v. Gompel and A. van den Bosch, “Efficient n-gram, skipgram and flexgram modelling with colibri core,” 2016.

[59] V. Gurusamy and S. Kannan, “Preprocessing techniques for text mining,” 10 2014.

[60] R. Mihalcea and P. Tarau, “TextRank: Bringing order into text,” in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, (Barcelona, Spain), pp. 404–411, Association for Computational Linguistics, July 2004.

[61] A. Kumar, A. Kumar, H. Hashmi, and S. A. Khan, “Wordpress: A multi-functional content management system,” in *2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART)*, pp. 158–161, IEEE, 2021.

Apêndice A

Comunicação entre os *Knowledge Services*

Os seguintes diagramas de sequência apresentam os *workflows* para processamento dos diferentes documentos jurídicos na solução desenvolvida. Nas figuras são representadas comunicações síncronas (através dos pedidos HTTP) e assíncronas (através do *Redis Streams*) entre os serviços.

De forma breve, o *content manager* insere um novo documento no sistema através de um pedido HTTP. O *orchestrator* implementa a lógica para processar cada documento, começando inicialmente em todos os casos por criar o nó que identifica o documento no grafo armazenado no *metadata broker*, através de um pedido HTTP.

Dependendo do documento a ser processado, um *knowledge service* pode ser ou não ser utilizado para extrair informação do *corpus* do documento. Por exemplo, o serviço que extrai informações do ELI só é executado durante o processo se for facultado o respetivo *link*.

Estas operações podem acontecer em paralelo face a independência que existe entre cada serviço presente na arquitetura. Para além disso, um serviço pode ser facilmente escalável. É importante que este processamento seja feito em conformidade com a documentação desenvolvida para garantir a integridade da informação na base de conhecimento e o correto funcionamento dos serviços.

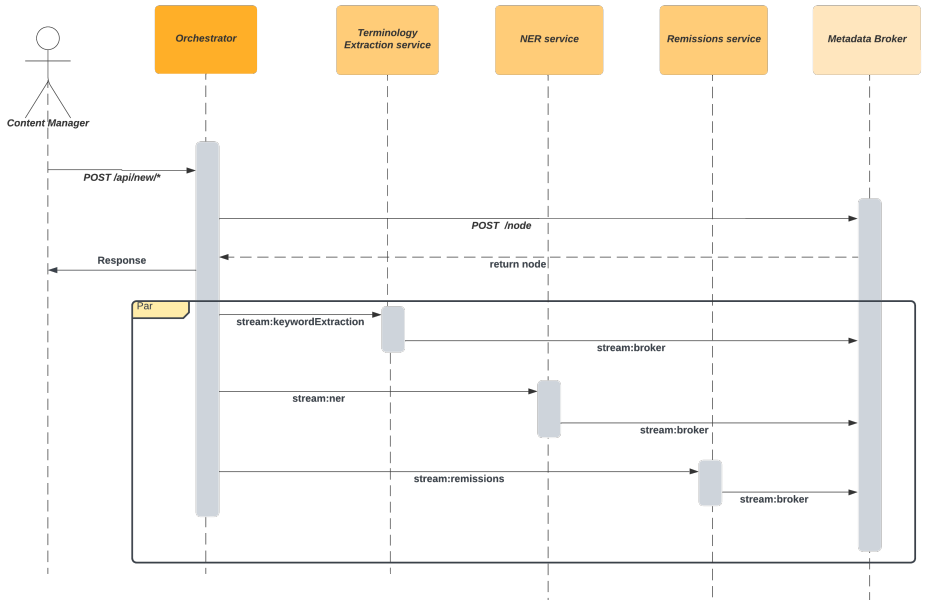


Figura 24: Diagrama de sequência para processar um caso prático, anotação ou artigo de opinião

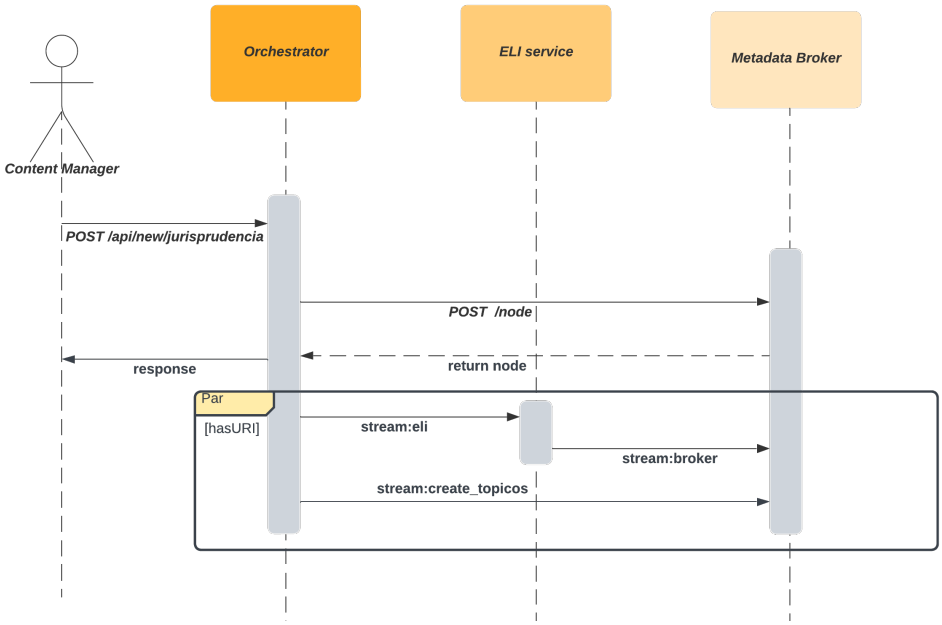


Figura 25: Diagrama de sequência para processar uma jurisprudência

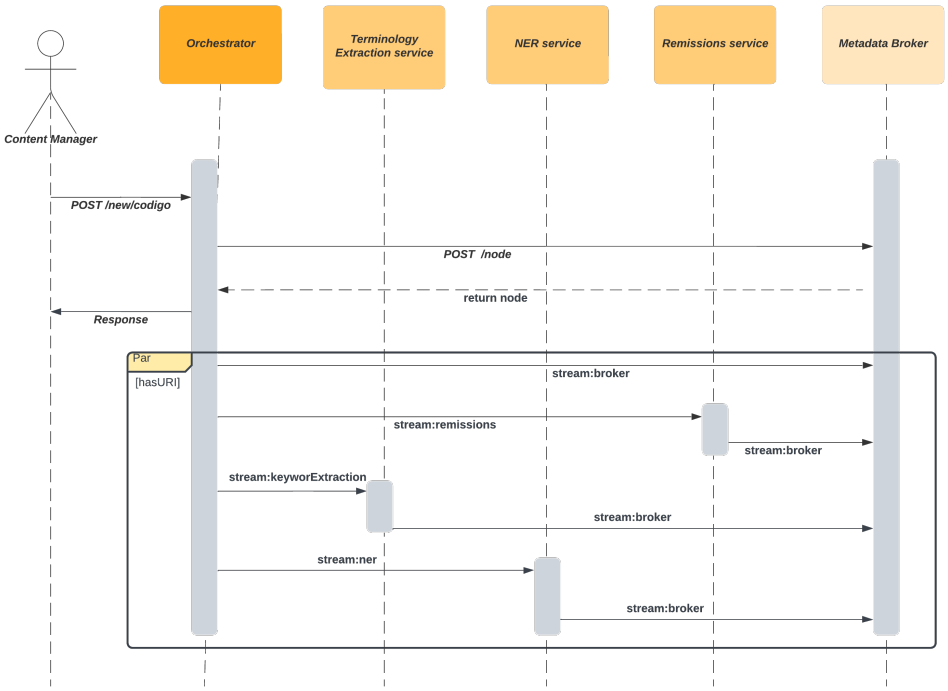


Figura 26: Diagrama de sequência para processar um código jurídico

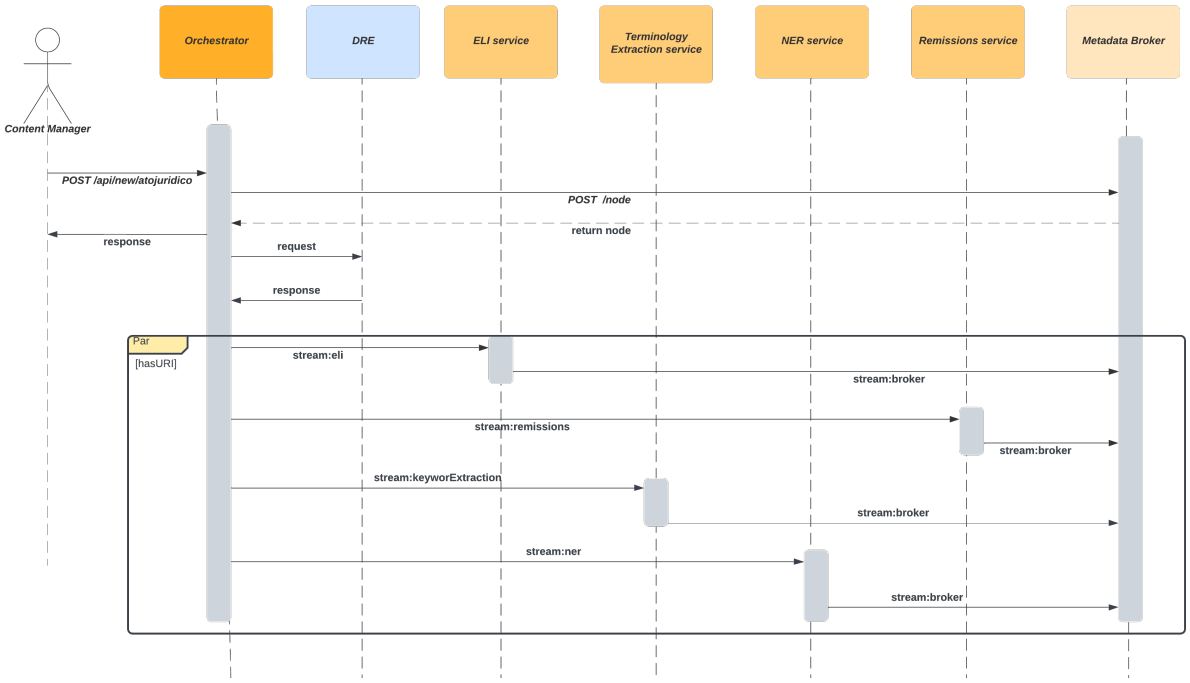


Figura 27: Diagrama de sequência para processar um ato jurídico