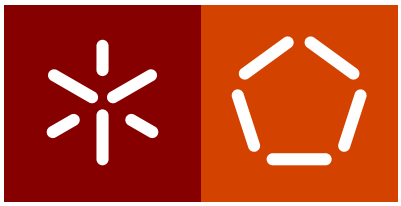
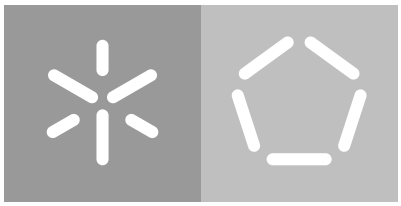


Universidade do Minho
Escola de Engenharia
Departamento de Informática

Tiago Miguel Fraga Santos

Anotação Automática de Textos para Análise e Identificação de Conteúdo

Julho 2022



Universidade do Minho
Escola de Engenharia
Departamento de Informática

Tiago Miguel Fraga Santos

Anotação Automática de Textos para Análise e Identificação de Conteúdo

Dissertação
Mestrado Integrado em Engenharia Informática

Dissertação supervisionada por
Professor Doutor Orlando Belo
Professora Doutora Anabela Barros

Julho 2022

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Atribuição-NãoComercial
CC BY-NC
https://creativecommons.org/licenses/by-nc/4.0/

AGRADECIMENTOS

Em primeiro lugar, quero agradecer à Universidade do Minho, ao Departamento de Informática e a todos os professores com quem me cruzei ao longo desta jornada, por toda a sabedoria, conhecimento e experiência que forneceram, pois sem eles não seria possível concluir esta etapa com sucesso.

Por toda a dedicação, empenho, prontidão, tempo e acima de tudo por toda a paciência, quero agradecer aos meus orientadores, o Professor Orlando Manuel Oliveira Belo e a Professora Anabela Leal Barros, pelo acompanhamento em todas as fases de desenvolvimento desta dissertação. Como orientando, foi um orgulho enorme poder realizar esta dissertação com o vosso apoio.

Ao Joel Morais, ao Reis, ao Chaves e ao Joel Peixoto por todos os momentos inesquecíveis durante a minha vida académica, pelo acompanhamento durante as aulas, durante os longos dias de estudo na biblioteca, e por todos os momentos de diversão durante esta etapa na minha vida.

À Daniela, à Sara e à Helena pela amizade, pelos momentos de diversão durante a minha vida académica, por todos os cafés e almoços que ficarão na minha memória.

Ao Dr.Agostinho por todas os conselhos, por todas as palavras amigas quando mais precisava e por toda a ajuda ao longo desta etapa.

À D.Rosa, ao Carlitos e ao Cau, por todo o carinho e por todos os momentos felizes ao longo destes anos, que me ajudaram a superar muitas dificuldades.

Ao João Gomes por toda a amizade desde o primeiro dia de aulas, por todos os trabalhos que fizemos em conjunto, por todos os dias de estudo, por todos os bons momentos que passamos juntos durante a nossa vida académica.

Ao meu Avô Manel e à minha Avó Maria, por tudo. Sem vocês não seria a pessoa que sou hoje. Muito obrigado por serem as minhas estrelinhas.

À minha namorada Raquel, por me acompanhar e apoiar incondicionalmente desde o início ao fim desta etapa. Por toda a ajuda nos momentos menos bons, e por todo o amor, carinho e compreensão que são fundamentais na minha vida.

À minha mãe que sempre me apoiou em todos os momentos da minha vida, por me proporcionar todas as condições necessárias ao meu sucesso, por me ajudar a ultrapassar todos os obstáculos, pela força incrível que sempre me deu, mas acima de tudo, por todo o amor.

ABSTRACT

Automatic text annotation systems are mechanisms that aim to provide assistance to users who need to extract and annotate relevant information in a given text. Usually, this type of system is developed for very specific application domains, in order to facilitate research processes on text content. The works of this dissertation will be developed based on the *Tombo da Mitra*, a codex that contains the inventory of the properties of the Archbishop's Table of Braga, in the 17th century. The quantity and diversity of the elements referred to in the book are impressive, as it contains all the names and surnames, settlements, professions, types of land and buildings, among many other elements, which are very important for the study and learning of geography, culture, economy, architecture, religion and portuguese language of the 17th century. The annotation of these elements expressively shows their location in time and space, as well as their potential relationships, facilitating the study of the book and providing linguistic researchers, teachers and students with a valuable instrument to reach and reinforce knowledge about the book. In this dissertation, we present a tool specially designed for the annotation of documents in the *Livro das Propriedades*, allowing the management and listing of annotation tags and providing a clearer view of the content of the manuscript.

Keywords: Annotation Systems, Automomatic Tagging, Data Analysis, Text Mining, Natural Language Processing, Machine Learning.

RESUMO

Os sistemas de anotação automática de textos são mecanismos que visam prestar auxílio a utilizadores que necessitem de extrair e anotar informação relevante num dado texto. Usualmente, este tipo de sistema é desenvolvido para domínios de aplicação bastante específicos, com vista a facilitar processos de pesquisa sobre conteúdos de textos. Os trabalhos da presente dissertação foram desenvolvidos com base no *Tombo da Mitra*, um códice que contém o inventário das propriedades da Mesa Arcebispal de Braga, no século XVII. A quantidade e diversidade dos elementos referidos no livro são impressionantes, uma vez que este contém nomes, apelidos, povoações, profissões, tipos de terrenos e edificações, entre tantos outros elementos, que são muito importantes para o estudo e aprendizagem da geografia, cultura, economia, arquitetura, religião e língua portuguesa até ao século XVII. A anotação destes elementos evidencia de forma expressiva a sua localização no tempo e no espaço, bem como as suas potenciais relações, facilitando o estudo do livro e proporcionando aos investigadores, linguistas, professores e alunos, um valioso instrumento para alcançar e reforçar o conhecimento sobre o manuscrito. Nesta dissertação, apresentamos uma ferramenta que foi concebida especialmente para a anotação dos documentos do Livro de Propriedades, que permite gerir e relacionar as etiquetas de anotação e proporcionar uma visão mais clara do conteúdo do referido manuscrito.

Palavras-Chave: Sistemas de Anotação, *Tagging* Automático, Análise de Dados, *Text Mining*, Processamento de Linguagem Natural, *Machine Learning*.

CONTEÚDO

LISTA DE FIGURAS

LISTA DE TABELAS

SIGLAS

API Application Programming Interface. 1, 9, 12

BPMN Business Process Model and Notation. 1, 8, 21

CSV Comma Separated Values Files. 1

GNU General Public License. 1

JSON JavaScript Object Notation. 1, 40

MIEI Mestrado Integrado em Engenharia Informática. 1

NER Name Entity Recognition. 1, 4, 12, 14, 15, 19–21, 55

PLN Processamento de Linguagem Natural. 1, 3, 4, 7, 9, 10, 12, 13, 15, 19, 20, 55, 57
POS Part of Speach. 1, 4, 14

UM Universidade do Minho. 1
UML Unified Modeling Language. 1, 35
URL Uniform Resource Locator. 1, 47

XLS Microsoft Excel Spreadsheet File. 1, 39
XML Extensible Markup Language. 1

1

INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Um sistema de anotação de texto é um mecanismo que identifica um conjunto de palavras-chave, geralmente designadas por *tags* (etiquetas), e que, de alguma forma, as associa aos elementos presentes num documento, de modo a que seja possível, posteriormente, identificar conceitos e padrões existentes nesse mesmo documento. Desta forma, um sistema de anotação auxilia os utilizadores que necessitem de extrair, anotar, indexar e pesquisar informação de um dado texto.

Usualmente, um sistema de anotação é desenvolvido para domínios de aplicação bastante específicos, com a função de facilitar os processos de pesquisa sobre conteúdos de textos. Apesar de a anotação de textos poder ser executada manualmente, a sua automatização é algo bastante desejado, uma vez que simplifica o processo de anotação e reduz drasticamente o seu tempo de realização. Além disso, permite estabelecer automaticamente todos os relacionamentos que possam envolver as *tags* descobertas (e anotadas) e os diversos elementos textuais contidos no documento analisado, facilitando posteriormente os processos de análise e melhorando o seu tempo de execução, eficácia e precisão.

Um sistema de anotação automático é construído através da utilização de mecanismos de PLN combinados com mecanismos de *machine learning*. Esta combinação de tecnologias permite-nos desenvolver sistemas que analisam os textos escritos em linguagem natural, identificando palavras e criando contextos de utilização, de acordo com um conjunto de pré requisitos estabelecido, que descobrem e mantêm as *tags* através de técnicas de processamento de texto, de forma mais expedita e com um nível de correção bastante elevado.

Estes sistemas são utilizados nos mais variados domínios, um pouco por todo o mundo. Por exemplo, a plataforma *Elketron* (Refinitiv, 2019) tem como objetivo fornecer aos seus utilizadores um sistema de anotação de texto para informações financeiras. Neste sistema, os elementos que se pretendem reconhecer e anotar são peças de informação sobre mercados financeiros, em que as *tags* descobertas (e os seus contextos) servirão para suportar decisões sobre as medidas a adotar em futuros investimentos. Outro exemplo interessante é a

plataforma *on-line Tagtop* (TagTop, 2019), que disponibiliza aos seus utilizadores meios para treinar um modelo de *machine learning*, permitindo suportar a criação de um sistema de anotação personalizado sobre uma dada área. Além destes exemplos, existem empresas de elevado estatuto mundial, como a *Uber*, a *Apple* ou a *Microsoft*, que utilizam vários procedimentos de anotação para analisarem a sua reputação no domínio das redes sociais.

1.2 Motivação e Objetivos

Nesta dissertação trabalhámos com base no *Tombo da Mitra*, um códice que contém o inventário das propriedades da Mesa Arcebispal de Braga no século XVII (Barros, 2019), (Barros, 2021) com o objetivo de desenvolver um sistema de anotação automática para os seus textos, que permitisse facilitar um processo de análise e de anotação da informação que eles contêm. Um leque significativo destes textos foram editados e posteriormente armazenados numa base de documentos do sistema *Tommi* (*tommi2.di.uminho.pt*).

O sistema *Tommi* foi desenvolvido por um grupo interdisciplinar, composto por cinco elementos da Universidade do Minho: dois professores, Anabela Barros, do Departamento de Estudos Portugueses e Lusófonos, e Orlando Belo, do Departamento de Informática, e quatro alunos do Mestrado Integrado em Engenharia Informática — João Gomes, Tiago Fraga, José Carvalho e Ricardo Martins. O sistema foi arquitetado e implementado de forma a que dispusesse de mecanismos para analisar, guardar e pesquisar a informação presente no códice do *Tombo da Mitra*.

O *Tombo da Mitra*, também designado como o *Livro das Propriedades*, incorpora um conjunto de 644 fólios, 1288 páginas manuscritas, de tamanho A3, contendo informação relativa aos tipos de terras, acidentes do terreno, nomes de ruas, proprietários e apontamentos biográficos e genealógicos, etc., relativos às propriedades da Mesa Arcebispal, que cruzavam todo o Minho e Trás-os-Montes no início do século XVII (Barros, 2019), (Barros, 2021).

Para que os processos de pesquisa e análise do conteúdo da base documental do sistema referido possam ser mais rápidos e efetivos, é necessário incorporar mecanismos de anotação de texto. A partir desta abordagem, é possível criar um conjunto de *tags* relevantes e indexadas, que permitirão descobrir graus de parentesco, árvores genealógicas, antropónimos, topónimos, tipos de terrenos, propriedades, entre outras. Deste modo, será possível manter uma base de *tags* como meio de indexação da informação mais relevante contida no *Tombo da Mitra*.

Adicionalmente, pretendeu-se também uma ferramenta que, com base numa dada especificação de uma *tag*, permitisse analisar os documentos contidos no sistema e, por similaridade, sugerir uma estratégia de anotação global para essa *tag*. Por fim, quisemos, também, desenvolver trabalho especificamente orientado para a criação e gestão de um mapa de relacionamentos de *tags* para descoberta de conteúdos semelhantes.

1.3 ESTRUTURA DA DISSERTAÇÃO

Para além do presente capítulo, esta dissertação está organizada em mais cinco capítulos. Assim, no segundo capítulo, será realizada a descrição de um sistema de anotação de texto, revelada a sua utilidade e especificidades únicas, de forma a demonstrar o desenvolvimento que tem vindo a ser feito nesse domínio. Será também apresentada uma análise comparativa de alguns sistemas de anotação, abordando alguns dos seus aspetos positivos e negativos. Ainda neste capítulo, serão descritas as ferramentas de PLN que foram estudadas, com o intuito de serem utilizadas como mecanismos de auxílio ao sistema de anotação que se pretendeu implementar. Este estudo suporta o início do desenvolvimento, uma vez que fornece uma perspetiva concreta sobre os componentes de processamento de texto, a sua utilidade e a forma como se definem os sistemas de anotação de texto.

No terceiro capítulo, será exposto o modelo conceptual do sistema de anotação desenvolvido. O capítulo começará por definir o modelo genérico de um sistema de anotação e de que forma ele se aplica aos sistemas existentes no mercado. Em seguida, será caracterizado o modelo adotado para a criação do sistema de anotação automática de textos.

No quarto capítulo, será descrito pormenorizadamente o caso de estudo abrangido nesta dissertação — o *Livro das Propriedades* e a aplicação *Tommi* —, e será demonstrada a necessidade de criar um sistema de anotação de textos no contexto estudado.

No quinto capítulo, será descrito o trabalho que foi desenvolvido ao longo desta dissertação, no qual se destacará o processo de atualização de grafia dos textos, que irão ser incorporados no sistema de anotação, bem como o processo que foi necessário para efetuar a anotação de cada uma das etiquetas. No final do capítulo descrever-se-á o processo de integração do sistema de anotação desenvolvido no sistema *Tommi*.

Por fim, no sexto e último capítulo desta dissertação, serão apresentadas algumas reflexões sobre o trabalho desenvolvido, abordando-se os pontos positivos e negativos, as dificuldades encontradas e as estratégias que foram adotadas para as mitigar. Este capítulo terminará com a apresentação de alguns comentários adicionais sobre a evolução do sistema de anotação desenvolvido nesta dissertação.

2

SISTEMAS DE ANOTAÇÃO DE TEXTO

2.1 DEFINIÇÃO

Um sistema de anotação de texto (Moraes and de Lima, 2008) é um mecanismo capaz de identificar conceitos e relações utilizando técnicas de mineração de texto, PLN e *machine learning*, com o intuito de recuperar e catalogar informação. Esta combinação de tecnologias permite desenvolver sistemas de análise de textos — escritos em linguagem natural — anotando palavras e os seus respetivos contextos de utilização. Os sistemas de anotação automática de textos permitem realizar a tarefa de anotação com um maior grau de rapidez e eficácia face à anotação manual.

Segundo Ferreira (2011), os sistemas de anotação automática de texto podem cumprir duas tarefas importantes. Em primeiro lugar, podem fazer o reconhecimento de entidades, um processo que se caracteriza por localizar e classificar elementos em textos de linguagem natural, segundo uma categoria pré-definida, como, por exemplo, localidades, países, nomes, profissões, entre outras. Em segundo lugar, este tipo de sistemas pode identificar pontos em comum entre entidades anotadas. Esta tarefa é importante em casos nos quais o objetivo seja criar uma ontologia de conceitos, de forma a entender as relações entre eles.

Usualmente, todos os sistemas de anotação de texto são compostos por conjuntos de ferramentas semelhantes (Ferreira, 2011), independentemente da área de aplicação. Deste modo, os sistemas de anotação contêm um *tokenizador*, que tem como objetivo dividir um texto em palavras (*tokens*), um detetor de limite de frase, que é capaz de extrair frases de texto, um marcador de Part of Speach (POS), que funciona como um analisador morfológico, pois identifica a classe gramatical de uma palavra (nome, verbo, adjetivo, etc.), um *gazetteer*, que é um dicionário que agrega os termos de uma determinada *tag*, e um Name Entity Recognition (NER). O NER é uma das ferramentas mais importantes dos sistemas de anotação. Este mecanismo caracteriza-se por ser capaz de identificar nomes próprios nos textos e classificá-los segundo conjuntos de categorias de interesse predefinidas, como, por exemplo, Pessoa, Local e Organização (Chu et al., 2012).

A extração de conceitos a partir de textos é fundamental em qualquer processo de recuperação de informação. Porém, este processo não é de fácil execução, pois a capacidade

de obter elementos anotados a partir de um grande conjunto de textos é, na verdade, uma tarefa bastante complicada (Chen et al., 2019).

2.2 Utilidade da Anotação de Texto

Nos últimos anos têm sido desenvolvidos bastantes trabalhos no campo da análise semântica de textos não estruturados, o que tem levado ao aparecimento de muitas aplicações em diversas áreas, como a interpretação de textos ou a recuperação e extração de conteúdos (Cornolti et al., 2013), (Chu et al., 2012).

Um dos grandes desafios deste tipo de ferramentas consiste no elevado número de documentos disponíveis nas empresas, nas bibliotecas, nas bases de dados ou até mesmo nas páginas *Web*. A partir deste grande volume de dados, a anotação manual de textos tornou-se uma tarefa demasiado dispendiosa em termos de tempo e dinheiro, uma vez que a anotação realizada por especialistas de uma determinada área se tornou lenta e pouco eficiente (Cai and Hofmann, 2003). Deste modo, a anotação automática de textos apresenta-se como uma solução viável para os problemas apresentados (Chu et al., 2012), contribuindo para a redução de recursos humanos, bem como para o aumento da taxa de eficácia no reconhecimento, na identificação e na extração de entidades em textos não estruturados.

Apesar de todas as dificuldades, o interesse em sistemas de anotação aumentou significativamente ao longo do tempo (Cornolti et al., 2013). Após a análise de alguns sistemas que surgiram mais recentemente, é possível afirmar que o crescimento da utilização destas ferramentas em diversas aplicações é notável, tome-se em consideração a área dos sistemas de *big data*. Por exemplo, os analistas de *sites* de *e-commerce* ou de redes sociais utilizam a anotação dos documentos do *site* para estabelecer padrões de comportamento do utilizador, assim como para descobrir tendências na aquisição de bens e serviços (Chen et al., 2019). Este aumento de interesse provocou, naturalmente, o aparecimento de várias iniciativas de investigação e de desenvolvimento de ferramentas de anotação, com particular relevância nos domínios de aplicação já mencionados.

2.3 Trabalhos Relacionados

Na área de extração e anotação de textos podemos encontrar vários sistemas capazes de operar em diversos domínios de estudo. Nesta secção iremos analisar alguns trabalhos que foram desenvolvidos em ambiente académico, bem como outros produtos de *software* existentes no mercado mundial.

2.3.1 *EXACT*

O primeiro exemplo a ser apresentado é a plataforma *EXACT* (Chen et al., 2019), que foi desenvolvida em 2019 por um grupo de colaboradores da Universidade de Zhejiang, na China. A plataforma *EXACT* (Entity eXtrACTion) foi implementada através do uso da *Framework Web Django* e da biblioteca de *Javascript React*. O sistema permite criar tarefas de anotação interativa, que criam índices de atributos e extraem entidades dos documentos usando esses índices.

O *EXACT* é capaz de extrair e anotar conteúdo de diferentes áreas de interesse. Estas áreas variam conforme os textos de entrada que são fornecidos ao sistema. Segundo os autores, a plataforma apresenta novidades a nível da conceptualização e implementação em relação a sistemas anteriormente desenvolvidos, pois o sistema integra ferramentas capazes de extrair entidades, fornecê-las a um sistema de *machine learning*, apresentar recomendações de anotação de texto em tempo real e, por fim, integrar estas técnicas numa interface para permitir a extração produtiva e eficiente. É através desta interface (Figura 1) que o utilizador assume um papel fundamental, uma vez que tem a função de validar as anotações sugeridas pelo sistema. Após a validação, a ferramenta de aprendizagem adquire novos dados para melhorar o algoritmo de anotação.

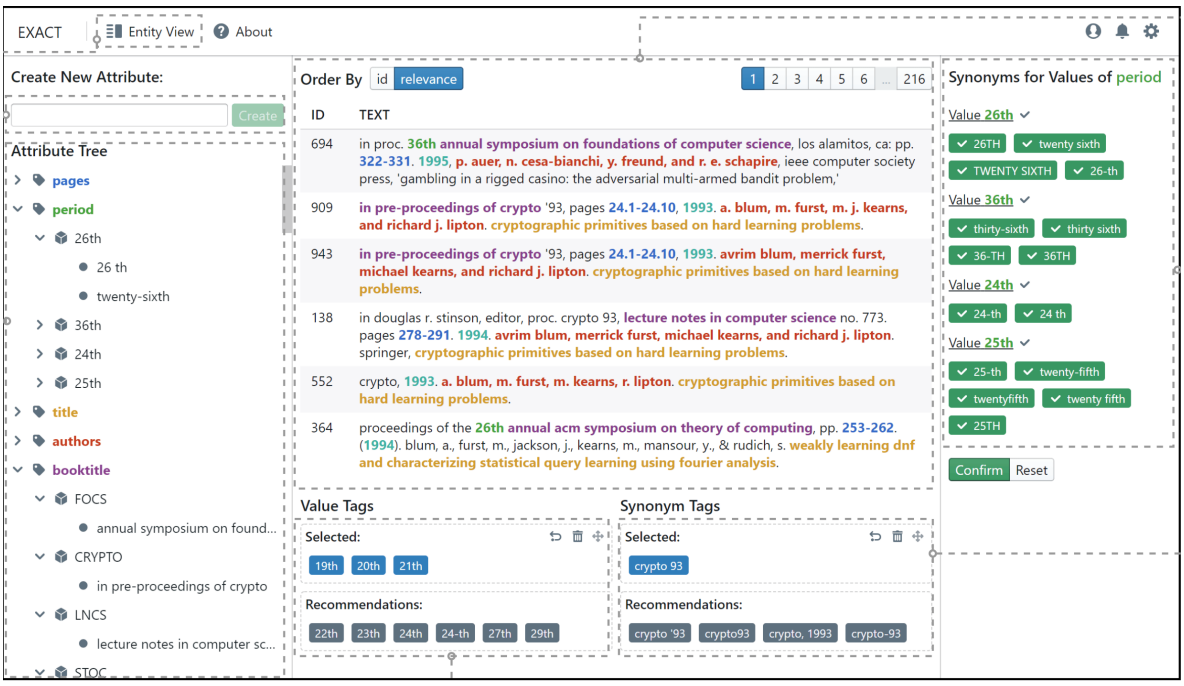


Figura 1: Interface do sistema *Exact*, retirado de Chen et al. (2019).

A ferramenta *EXACT* é composta por quatro componentes principais: 1) o módulo de etiquetação, que executa as tarefas de anotação das palavras; 2) o módulo de indexação, que

guarda num dicionário a informação fornecida pelo módulo de etiquetação; 3) o módulo de recomendação, que está encarregue de sugerir novas etiquetas de anotação; por fim, 4) o módulo de criação de novas entidades anotadas.

Através das características enumeradas, é possível afirmar que esta ferramenta possui componentes e arquiteturas semelhantes ao sistema que está a ser desenvolvido no âmbito dos trabalhos desta dissertação.

2.3.2 *Elketron*

Com o passar dos anos, têm vindo a ser desenvolvidos outros sistemas de anotação de texto pertencentes a áreas de interesse distintas. Na área da economia e dos mercados financeiros é possível destacar a plataforma *Elketron* (Refinitiv, 2019).

A *Refinitiv*[1] é uma empresa tecnológica que tem como principal modelo de negócio o fornecimento de dados sobre a área financeira. Esta empresa, situada em Times Square, Nova Iorque, desenvolveu a plataforma *Elketron Data Platform* com o objetivo de extrair e anotar notícias que possam interferir com as ações em bolsa dos mercados financeiros. A plataforma é bastante usada por bancos a nível mundial, como é o caso do *Standard Chartered*. No entanto, antes de os dados estarem acessíveis, é necessário fazer um tratamento dos mesmos.

Segundo a *Refinitiv*, o componente responsável pela anotação inteligente (Refinitiv, 2020) utiliza técnicas de PLN, análise de texto e mineração de dados para derivar o significado de grandes quantidades de conteúdo não estruturado, uma vez que, segundo a empresa, é a maneira mais rápida, fácil e precisa de marcar pessoas, lugares, factos e eventos nos mais variados dados. O componente de anotação inteligente oferece suporte a textos escritos em inglês. No entanto, esta ferramenta também suporta alguns conceitos-chave de outros idiomas, tais como o mandarim, o japonês, o alemão, o espanhol e o francês.

No seu *site*[2], a *Refinitiv* permite que sejam efetuadas demonstrações do seu sistema de anotação inteligente. Na Figura 2 é possível analisar o resultado do trabalho de anotação efetuado pelo componente em questão. No exemplo apresentado foi utilizada uma notícia de um jornal britânico sobre a COVID-19[3].

1 https://www.refinitiv.com/
2 https://permid.org/onecalaisViewer
3 https://www.dailymail.co.uk/news/article-9380039/Britains-vaccine-shortage-April-WONT-hamper-inoculation-drive-says-Prof-Lockdown.html

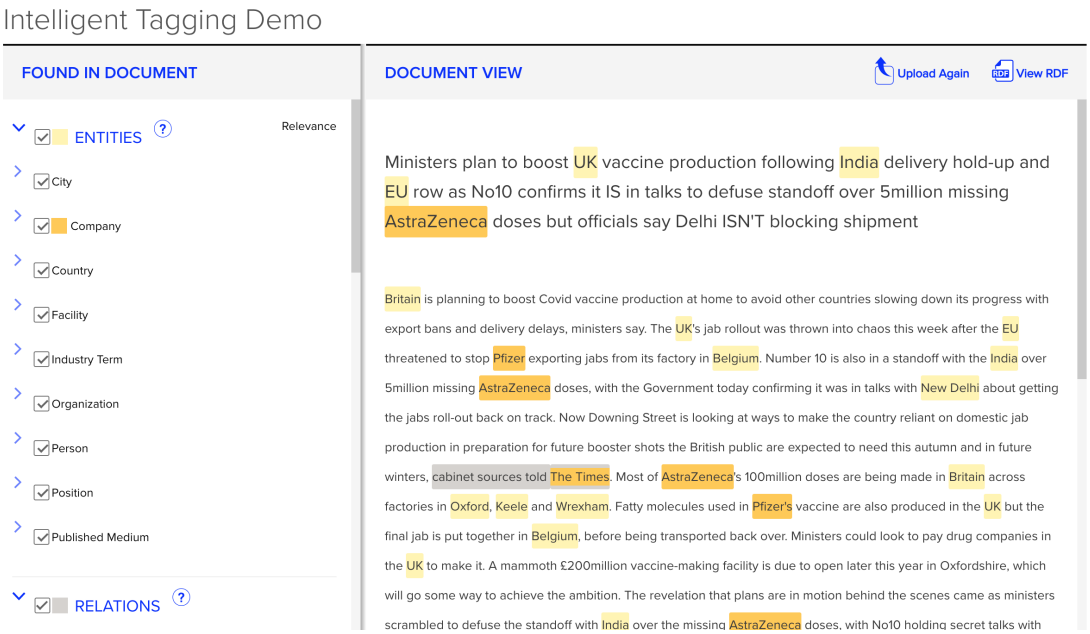


Figura 2: Anotação de notícia sobre a COVID-19.

Além da área da economia, a *Refinitiv* decidiu utilizar o seu mecanismo de anotação inteligente noutros modelos de negócio, como, por exemplo, o jornalismo. Um dos clientes da *Refinitiv* nesta área é a *NBC News*. Apesar de ser uma empresa gigante no ramo do jornalismo, a *NBC News* realizava o processo de anotação e recuperação de informação manualmente, o que levava a vários erros e tornava a tarefa morosa. Desta forma, a *NBC News* procurou abandonar o processo manual e encontrar uma solução que automatizasse estas tarefas. Nesse sentido, a *NBC* adotou o modelo de anotação inteligente da *Refinitiv*, que é capaz de processar rapidamente textos com conteúdo não estruturado, como notícias, relatórios de pesquisa, redes sociais ou *blogs*, de forma a localizar entidades, relacionamentos, factos e eventos (Refinitiv, 2021). Em suma, segundo os casos de estudo apresentados nesta secção, a *Refinitiv* conseguiu desenvolver um sistema de anotação automática de texto multifuncional capaz de operar em diferentes áreas de negócio.

2.3.3 *Tag Top*

A plataforma TagTop (2019) é uma ferramenta *open-source* de anotação automática de textos que pode ser utilizada a partir de um servidor remoto ou através de uma instalação local numa máquina específica. Esta ferramenta fornece uma interface (Figura 3) na qual o utilizador é capaz de anotar entidades em documentos textuais. Segundo os autores do sistema, a *TagTop* fornece uma interface adaptável que permite ao utilizador visualizar apenas as ferramentas de que necessita para a tarefa de anotação. Além da capacidade de anotação manual, a ferramenta disponibiliza meios para a anotação automática de textos,

com o objetivo de aumentar a produtividade da tarefa. A anotação automática é conseguida através da utilização de entidades anotadas manualmente, ou de dicionários, e de modelos de *machine learning*. Estes últimos podem ser da própria ferramenta ou modelos externos importados pelo utilizador.

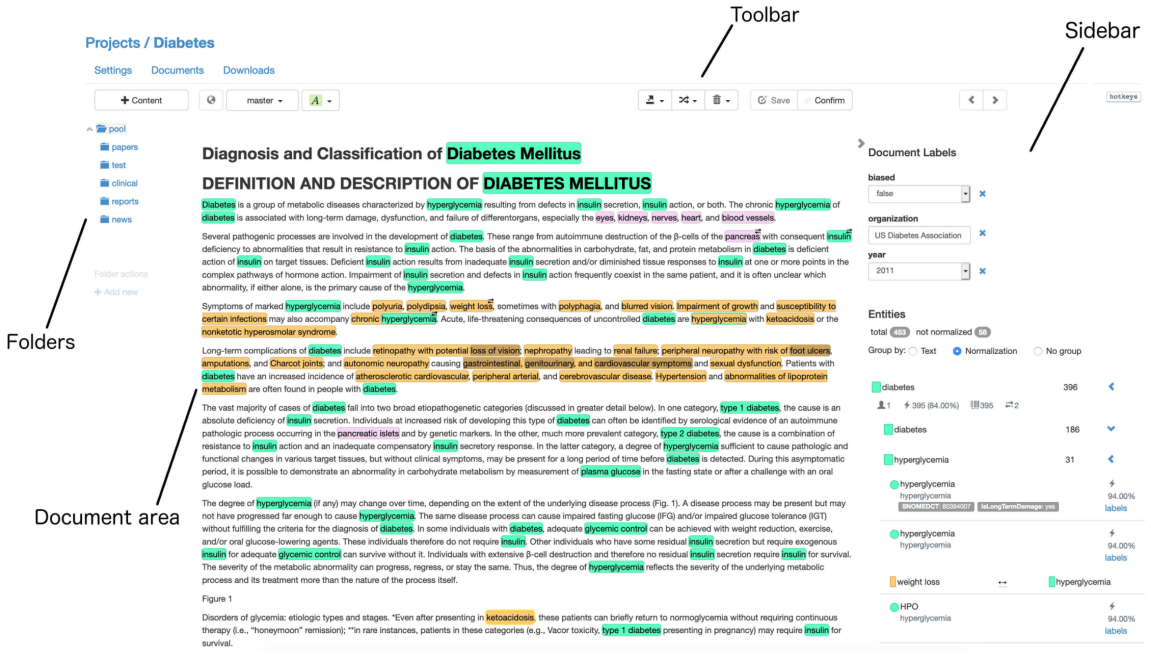


Figura 3: Exemplo de anotação através do *TagTop*.

A *TagTop* foi desenvolvida com o auxílio de algoritmos de PLN. Esta utiliza as correções manuais efetuadas sobre a anotação automática para melhorar a precisão dos seus modelos. Assim como em outros sistemas de anotação de texto similares, a *TagTop* foi desenvolvida a partir da integração de algoritmos de aprendizagem semi-supervisionada. Este tipo de algoritmos usa uma grande quantidade de dados não anotados em consonância com os dados manualmente anotados, de forma a construir modelos de aprendizagem mais robustos e precisos. Por outro lado, um algoritmo não supervisionado apenas utiliza dados não anotados, o que torna mais difícil atingir níveis de precisão elevados. Além disso, é uma alternativa que se revela mais dispendiosa.

Por fim, os criadores de *software*, por exemplo, podem tirar também partido da capacidade de anotação da *TagTop*, usando a Application Programming Interface (API) disponibilizada pela plataforma[4]. Com a integração deste sistema numa ferramenta construída de raiz, a API da *TagTop* oferece métodos para efetuar a autenticação no sistema e para importar e anotar documentos simples ou anotados previamente. A integração pode ser efetuada recorrendo às linguagens *JavaScript*, *Python* ou através do comando *Curl*.

4 https://docs.tagtog.net/API_documents_v1.html

2.3.4 *AdaBoost*

Na área de extração e classificação de conhecimento, também é possível estudar algoritmos de *machine learning*, como é o caso do mecanismo *Adaboost*. Este mecanismo, proposto por Cai and Hofmann (2003), utiliza duas fases de aprendizagem. A primeira fase é não supervisionada, pois tem o objetivo de extrair de forma automática conceitos de uma dada área de estudo. Os conceitos extraídos na primeira fase são processados numa segunda fase que é, por sua vez, uma etapa supervisionada. O *Adaboost* possui a flexibilidade necessária para ser integrado num sistema de anotação de texto que venha a ser desenvolvido. Assim, a partir da sua integração, as ferramentas de *machine learning* inerentes a qualquer sistema de extração de conteúdo poderão ser melhoradas.

Este algoritmo provou ser bastante versátil em termos de integração com sistemas externos, uma vez que, segundo Bloehdorn et al. (2006), foi utilizado no desenvolvimento de um sistema de anotação de texto suportado por bases de dados e ontologias externas, de forma a facilitar a identificação de conceitos-chave na anotação de textos. O sistema desenvolvido por Bloehdorn et al. (2006) é capaz de extrair e anotar entidades e de melhorar as ontologias de uma forma automática.

2.3.5 *CLUTO*

O *CLUTO* (Karypis, 2003) é outro sistema que pode ser utilizado para fazer a extração de conceitos de documentos textuais. No estudo realizado por Moraes and de Lima (2008) apresenta-se como um sistema capaz de extrair e anotar conceitos, que dispõe de algoritmos de agrupamento de dados.

Durante os testes do sistema, foram utilizados textos em língua portuguesa, enquanto a maioria dos trabalhos realizados neste domínio executaram esta tarefa com textos em língua inglesa. Segundo esses autores, estruturas conceptuais como dicionários, *thesaurus*[5], taxinomias[6] e ontologias têm-se tornado recursos importantes em sistemas de informação, uma vez que estas estruturas têm obtido excelentes resultados em termos de precisão na identificação de conceitos. Apesar da aplicação destas estruturas de dados em várias áreas, a dificuldade de desenvolver e manter um sistema deste tipo é bastante elevada.

Com o intuito de construir um sistema robusto e preciso, foram utilizados textos de desporto, devido à simplicidade do vocabulário que estes oferecem, quando comparados com os textos de outras áreas, como a literatura ou a história. Numa fase inicial, os

5 Um thesaurus é um dicionário que regista uma lista de palavras que são associadas semanticamente a outras, apresentando geralmente sinónimos e, algumas vezes, antónimos.

6 Uma taxionomia é conjunto de princípios e métodos de classificação dos diversos elementos de uma área científica, isto é, um sistema de categorização.

textos foram pré-processados, tendo as palavras e os sinais de pontuação sido separados e etiquetados morfologicamente, com uma precisão próxima dos 95%.

O desenvolvimento do sistema de anotação e de extração de conhecimento de Moraes and de Lima (2008) baseou-se na utilização de dependências sintáticas entre os verbos e os seus argumentos, com o objetivo de identificar os termos e multi-termos (n-gramas[7]) mais relevantes ao longo de um texto. Neste sistema, os conceitos são obtidos e organizados através de algoritmos de agrupamento que estão disponíveis na ferramenta *CLUTO*.

Apesar da utilização da ferramenta *CLUTO* ter-se mostrado adequada na extração e anotação de termos, os autores detetaram alguns problemas na geração dos *clusters* de determinados conceitos, podendo significar que a integração do *CLUTO* com um sistema desenvolvido de raiz possa originar algumas dificuldades.

2.3.6 *Análise Comparativa*

Nesta secção irá ser apresentada uma análise comparativa entre os mecanismos de anotação de texto que foram descritos ao longo da Secção 2.3. Para a realização desta análise foi necessário estabelecer alguns pontos de comparação entre os sistemas, que são elementos fundamentais em qualquer sistema de anotação automática de textos. No total, foram considerados três atributos fulcrais para um sistema de anotação de texto, cujo principais objetivos são a extração de conhecimento de um documento textual não estruturado e a fácil integração em sistemas externos.

Em primeiro lugar, foi considerado o atributo "Integrável", que caracteriza uma ferramenta de anotação de texto como sendo capaz de ser integrada noutras ferramentas ou mecanismos externos. Em segundo lugar, foi considerado o atributo "Utilizável", que caracteriza uma ferramenta de anotação de texto capaz de ser utilizada na extração de conceitos de um texto, sem a necessidade de qualquer desenvolvimento adicional. Por último, o atributo "Aprendizagem", que caracteriza uma ferramenta de anotação de texto capaz de melhorar a precisão da anotação a partir de iterações executadas anteriormente.

	Integrável	**Utilizável**	**Aprendizagem**
EXACT	Não	Sim	Sim
Elketron	Não	Sim	Sim
Tag Top	Sim	Sim	Sim
AdaBoost	Sim	Não	Sim
CLUTO	Sim	Não	Sim

Tabela 1: Comparação entre Mecanismos de Anotação de Texto.

7 No campo da linguística computacional, um n-grama é uma sequência contígua de n itens de uma determinada amostra de texto ou fala. Os itens podem ser fonemas, sílabas, letras, palavras ou pares de bases de acordo com a aplicação.

Através da Tabela 1 é possível fazer uma comparação entre os mecanismos de anotação de texto estudados, tendo em atenção os atributos considerados para esta análise. Numa primeira instância, verifica-se que todos os mecanismos estudados têm a capacidade de aprender e melhorar a precisão de anotação, consoante a execução de diferentes iterações. Esta característica indica que, ao longo do tempo, a precisão da anotação de conceitos num documento não estruturado tende a aumentar.

Os sistemas *EXACT* e *Elketron* são duas ferramentas capazes de serem utilizadas na extração de conceitos. No entanto, estas ferramentas não possuem a capacidade de se integrarem em sistemas ou aplicações externas. O *EXACT* foi desenvolvido de raiz, de forma a oferecer ao utilizador final um sistema de anotação e extração de conceitos, através de uma interface simples e intuitiva. Do mesmo modo, o *Elketron*, por ser um produto cuja propriedade intelectual pertence à empresa *Refinitiv*, não é suscetível de ser integrado em ferramentas externas de uma forma livre e gratuita.

Quanto aos sistemas *AdaBoost* e *CLUTO*, apesar de poderem ser integrados em sistemas externos de uma forma *open-source*, não são ferramentas passíveis de serem imediatamente utilizadas sem desenvolvimentos adicionais. Os principais componentes destas ferramentas são algoritmos de *machine learning*. Portanto, para cumprirem as tarefas de anotação e extração de conhecimento de um dado documento, devem ser agregadas a um sistema externo, cuja principal função seja a anotação de conceitos.

Por fim, observa-se que o sistema *Tag Top* possui todos os atributos abordados, uma vez que é integrável, utilizável e tem a capacidade de aprendizagem ao longo das diferentes iterações. Esta ferramenta obedece ao atributo "Utilizável", pois pode ser imediatamente usada na tarefa de anotação de textos e, ainda, possui a característica "Integrável", já que oferece uma API *open-source* para os criadores de *software* utilizarem no desenvolvimento de sistemas de anotação.

Em suma, os sistemas *Tag Top* e *EXACT* são aqueles cujas características mais se aproximam do sistema de anotação automática de textos desenvolvido no âmbito da presente dissertação.

2.4 Ferramentas de Processamento de Linguagem Natural

Após o estudo dos sistemas de anotação atualmente existentes no mercado, compreendeu-se que era importante integrar no sistema uma ferramenta que permitisse apoiar o processo de separação e classificação de palavras nos textos. Para extrair conhecimento útil a partir de documentos escritos em língua portuguesa, é essencial ter a capacidade de reconhecer nomes, verbos, pronomes, adjetivos, etc. Para a execução dessa tarefa, foram estudadas algumas ferramentas de PLN, cuja principal função fosse o NER. O NER é uma sub-tarefa da

extração de conhecimento e refere-se à capacidade de reconhecer nomes de pessoas, locais e organizações em documentos de língua natural (Ferreira, 2011), (Chu et al., 2012).

Nas próximas secções irão ser descritas as duas ferramentas estudadas com capacidade para desempenhar as funções acima mencionadas.

2.4.1 *LinguaKit*

A primeira ferramenta estudada foi o *LinguaKit* (Gamallo and Garcia, 2017). Esta é uma ferramenta de PLN que contém módulos de análise, de extração e de anotação linguística. O *LinguaKit* foi desenvolvido com a linguagem *Perl*[8], na Universidade de Santiago de Compostela, sendo capaz de suportar quatro linguagens: o galego, o espanhol, o português e o inglês. Além disso, o serviço está disponibilizado como uma aplicação *web*[9] e o código encontra-se disponível em formato *open-source*[10].

O *LinguaKit* encontra-se dividido em quatro módulos. O primeiro módulo é composto por ferramentas capazes de identificar e conjugar verbos e realizar a tradução de textos. O segundo módulo está orientado para utilizadores que pretendam realizar estudos no âmbito educacional, uma vez que esta ferramenta possui um analisador lexical e morfossintático. Por sua vez, o terceiro módulo destina-se a profissionais de comunicação e *marketing*, pois possui mecanismos capazes de realizar a análise de sentimentos de um determinado texto. Por último, o *LinguaKit* possui um módulo experimental, no qual é possível testar as novas ferramentas do projeto. Nestes quatro módulos, são apresentadas dezasseis componentes que podem ser invocadas e usadas externamente.

Perante a facilidade de integração desta ferramenta nos trabalhos em curso e face à vasta gama de funcionalidades que apresenta, foi decidido incorporar o *LinguaKit* nos trabalhos da presente dissertação, de forma a automatizar os processos de separação e de categorização de palavras. Para tal, dos dezasseis componentes disponíveis na ferramenta, apenas dois foram utilizados: o *tokenizer* e o *tagger*.

O primeiro componente, o *tokenizer*, possui a capacidade de segmentar o texto em frases e as frases em palavras. Este processamento é realizado a partir da identificação das fronteiras de cada uma das frases, com base em máquinas de estado finitas e em listas de sinais de pontuação. O segundo componente, o *tagger*, é responsável por reconhecer a classe morfológica de cada uma das palavras. Os autores avaliaram o funcionamento deste módulo e conseguiram resultados com precisão próxima dos 96% para documentos redigidos em língua portuguesa e espanhola, e 94% para documentos escritos na língua inglesa.

8 https://www.perl.org/
9 https://www.linguakit.com
10 https://github.com/citiususc/Linguakit

2.4.2 *FreeLing*

O *FreeLing* (Padró and Stanilovsky, 2012), desenvolvido no centro de pesquisa TALP2, foi a segunda ferramenta estudada, com o objetivo de ser integrada nos trabalhos da presente dissertação. Esta ferramenta encontra-se num formato *open-source* e foi desenvolvida na linguagem *C++*.

A arquitetura do sistema consiste numa abordagem simples de cliente-servidor, pois possui uma camada para o processamento linguístico e uma camada de interface que solicita ao utilizador os resultados conforme o objetivo da aplicação.

Este sistema tem uma vasta lista de funcionalidades de análise de texto, entre as quais é possível destacar a análise morfológica, a deteção de entidades (NER), a marcação POS (*tagger*), a desambiguação do sentido da palavra, a rotulagem de função semântica, a identificação da linguagem do texto e a separação de palavras (*tokenizer*), entre outras. Contudo, à semelhança do que aconteceu com o *FreeLing*, nem todos os componentes foram considerados para os trabalhos desta dissertação. Apenas o NER, o *tagger* e o *tokenizer* foram utilizados.

Segundo os autores, o componente NER necessita de dois módulos externos para realizar a tarefa: um módulo de capitalização (que converte letras maiúsculas em minúsculas e vice-versa) e um módulo capaz de reconhecer entidades. Este componente consegue atingir uma precisão a rondar os 90%. Por sua vez, o *tagger* é capaz de receber uma lista de frases e anota cada uma das palavras com o POS, atingindo uma precisão de 97%-98%.

A versão 3.0 do *FreeLing* apresenta várias alterações pertinentes, de modo a tornar a ferramenta mais prática e flexível. As mudanças desta versão podem-se agrupar em três tipos: mudanças relacionadas com a capacidade multilingue, uma vez que foram adicionadas novas línguas ao sistema; modificações nos componentes de *machine learning*, de forma a melhorar a capacidade de aprendizagem com as iterações anteriores, e ainda alterações relacionadas com os aspetos de engenharia do projeto.

Atualmente esta versão encontra-se a ser utilizada por diversos projetos industriais, entre os quais é de destacar o *Ruby Reader* [11], o *Vi-Clone* [12], o *TextToSign* [13], o *Dixio* [14] e o *Aport News* [15].

O *FreeLing* encontra-se disponível em vários idiomas, nomeadamente português, espanhol, catalão, galego, inglês, italiano, francês, alemão, russo, croata e esloveno.

11 http://www.camobile.com)
12 http://www.vi-clone.com
13 http://www.textosign.es
14 http://www.semantix.com
15 http://news.aport.ru

2.4.3 *Análise Comparativa*

O estudo das ferramentas de PLN permitiu fazer uma avaliação sobre qual possui as melhores características para ser integrada nos trabalhos desenvolvidos. Como mencionado anteriormente (Secção 2.4), o sistema de anotação deve ter a capacidade de reconhecer entidades nomeadas. Ora, pelo estudo efetuado, verificou-se que, através do uso de componentes como o *tokenizer*, o *tagger* e o NER — que ambas as ferramentas possuem — era possível realizar esta tarefa com sucesso.

Relativamente às ferramentas apresentadas, pode-se concluir que tanto o *LinguaKit* como o *FreeLing* são ferramentas muito semelhantes em termos das componentes que oferecem, como constatado pela Tabela 2.

Componente	LinguaKit	FreeLing
Conjugador verbal	Sim	Sim
Segmentador de orações	Sim	Sim
Tokenizer	Sim	Sim
Lematizador	Sim	Sim
PoS tagger	Sim	Sim
Identificador de entidades	Sim	Sim
Classificador de entidades	Sim	Sim
Identificador de co-referência	Sim	Sim
Analisador sintáctico em dependências	Sim	Sim
Expressões multi-palavra	Sim	Sim
Analisador de sentimentos	Sim	Sim
Sumarização	Sim	Não
Anotação semântica	Sim	Sim
Identificação de linguagem	Sim	Sim
Correcção/Avaliação linguística	Sim	Não

Tabela 2: Comparação entre ferramentas de PLN.

Visto que estes dois sistemas foram desenvolvidos em contexto académico, pode-se afirmar que a maior diferença entre eles é o número de línguas que cada um suporta. Neste campo, o *FreeLing* encontra-se em vantagem, pois além do português, do inglês, do espanhol e do galego, consegue também suportar o francês, o italiano, o catalão, o russo, o alemão, o croata e o esloveno. Todavia, este não foi um fator preponderante para a nossa escolha, uma vez que os textos processados no âmbito desta dissertação se encontram-se redigidos apenas em língua portuguesa.

A integração destas ferramentas é realizada através de linha de comandos, no entanto, foram encontradas várias dificuldades no momento de implementar o *FreeLing* com o sistema de anotação desenvolvido. Por outro lado, a integração do *LinguaKit* foi efetuada de uma forma muito mais fácil, intuitiva e rápida.

Durante o estudo destas ferramentas, foi encontrada uma diferença ao nível do analisador morfossintático. O *LinguaKit* apresenta divisões das preposições[16] no resultado final, enquanto que o *FreeLing* não revela este comportamento. No entanto, esta situação não foi considerada relevante pois pode ser facilmente contornada e, ao mesmo tempo, fornece um maior nível de detalhe à anotação.

Em suma, é possível considerar que nenhuma das ferramentas se destaca, uma vez que ambas foram desenvolvidas sobre modelos bastante semelhantes e oferecem praticamente a mesma quantidade de componentes e soluções. Para os trabalhos da presente dissertação, foi então decidido integrar o *LinguaKit* em detrimento do *FreeLing*, essencialmente devido ao grau de familiaridade com a ferramenta e à maior facilidade de implementação, pois funciona como programa executável em linha de comandos, uma característica que se revelou fundamental.

16 Por exemplo: A preposição ”da”é dividida em ”de”e ”a”.

3

O SISTEMA DE ANOTAÇÃO AUTOMÁTICA DE TEXTOS

3.1 ANOTAÇÃO DE TEXTOS

Um texto é um conjunto de orações que transmitem algum tipo de informação ao leitor. No entanto, se não existirem marcações ou notas ao longo do texto, a sua interpretação pode sofrer algumas diferenças entre diferentes leitores, ou seja, dentro de um texto, uma determinada frase pode transmitir uma ideia a um leitor e a outro uma ideia completamente diferente.

A anotação de textos (Donnell and Donnell, 2004) é uma tarefa que adiciona valor aos textos, uma vez que, a partir da utilização de um conjunto bem definido de marcas (*tags* ou etiquetas), é possível ajudar o leitor na sua interpretação. Por exemplo, ao analisar um texto convencional encontram-se com grande facilidade inúmeros sinais de pontuação — correspondendo basicamente a um conjunto de marcas — que têm como função ajudar na leitura e na interpretação desse mesmo texto.

As etiquetas, definidas de acordo com o conteúdo dos textos, são denominadas como "anotações orientadas para o conteúdo"(Ferreira, 2011) e pretendem indicar a presença de diferentes tipos de elementos no texto, tais como nomes (antropónimos), lugares (topónimos), profissões, ou produtos, entre outros. Quanto maior a presença dessas etiquetas, maior será a percentagem de texto anotado e, consequentemente, mais fácil será a sua interpretação.

Frequentemente, a anotação de textos é realizada de forma *ad hoc*, o que, obviamente, não transmite ao leitor a verdadeira utilidade desta técnica. No entanto, quando as anotações são feitas de forma metódica e orientada, a capacidade de compreensão e de análise dos textos aumenta de forma significativa, a partir da evidenciação das principais ideias do texto. Desta forma, esta técnica pode ajudar o leitor a expressar as suas próprias ideias e pensamentos, permitindo um acesso rápido e direto aos elementos mais pertinentes de cada texto (Erin Lynch, 2021).

3.2 MODELOS E PROCESSOS DE ANOTAÇÃO

Nos últimos anos, foram desenvolvidas diversas aplicações no âmbito da extração de conhecimento e anotação automática de textos. A criação destes sistemas ocorreu tanto em ambiente académico como em ambiente industrial. Neste último caso, algumas aplicações já atingiram um grau de maturidade bastante elevado. Veja-se, por exemplo, a aplicação *EXACT* (Chen et al., 2019), que foi desenvolvida em contexto académico na Universidade de Zhejiang, na China, e que tem aplicação em diversas áreas. Também a plataforma *Elketron* (Refinitiv, 2019) foi desenvolvida em contexto industrial pela *Refinitiv*, tendo como principal objetivo a anotação de textos nas áreas da economia e dos mercados financeiros, como já atrás foi referido.

Independentemente do contexto de desenvolvimento e da área de aplicação, a boa prática de criação de *software* recomenda a definição de um esquema de anotação robusto, que defina claramente como anotar o texto ou o tipo de *tags* a utilizar. Além de um bom esquema de anotação, esta tarefa requer o envolvimento de muitas pessoas e, em particular, de ferramentas especializadas que forneçam funcionalidades de reconhecimento de entidades e de divisão de palavras. A anotação de texto não é uma tarefa simples (Finlayson and Erjavec, 2017).

Com o objetivo de estudar e compreender a arquitetura de diferentes sistemas de anotação, foram analisados diversos trabalhos para que pudéssemos analisar aquilo que de melhor cada modelo desenvolvido podia proporcionar. Por exemplo, no sistema de anotação elaborado por Dias et al. (2020) foi implementado um modelo híbrido capaz de reconhecer e classificar entidades a partir de textos não estruturados escritos em português europeu. Este modelo (Figura 4) combinou a utilização de várias técnicas, como, por exemplo, a segmentação de palavras, a análise morfológica, os modelos de *machine learning* e os modelos baseados em regras lexicais.

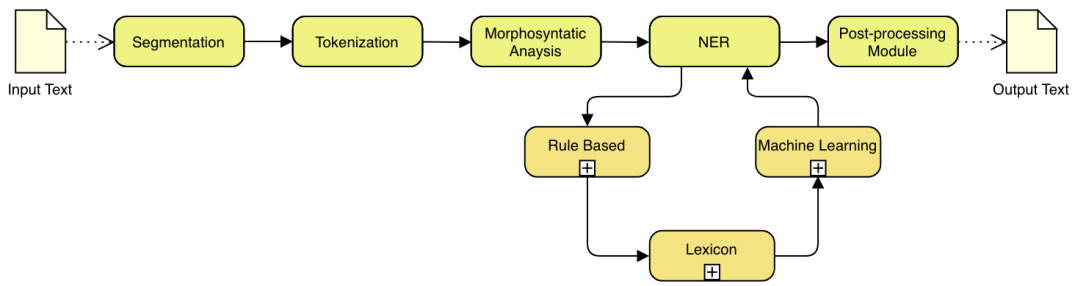


Figura 4: Modelo do sistema de anotação — figura adaptada de Dias et al. (2020).

No desenvolvimento do sistema de anotação foi adotada uma arquitetura modular. Dessa forma, os diferentes módulos do sistema podem ser configurados separadamente e executados várias vezes de forma independente. O primeiro módulo do sistema de anotação de

Dias et al. (2020) corresponde ao pré-processamento e ao tratamento dos textos, sendo o PLN uma das tarefas mais importantes. O pré-processamento encontra-se dividido em três fases: a segmentação de frases, que é capaz de extrair as frases de um texto, a separação de orações (*Tokenization*) que divide o texto em elementos lexicais (*tokens*), e a análise morfológica, que consiste na classificação das palavras consoante a sua classe morfológica (nome, verbo, adjetivo, etc.). Assim que termina a execução deste módulo, o texto processado é fornecido ao segundo módulo do sistema: o NER. É neste módulo que são implementados os modelos baseados em regras lexicais e de *machine learning*.

O *GermaNER* (Benikova et al., 2010) é um sistema de anotação de textos que foi desenvolvido com o objetivo de reconhecer e classificar entidades em textos de língua alemã. Segundo os autores, podem existir diferentes abordagens no desenvolvimento de sistemas NER. A modelação destes sistemas pode incorporar uma abordagem baseada em regras lexicais e em algoritmos de *machine learning*, supervisionados ou semi-supervisionados. Apesar de uma modelação a partir de regras lexicais (ou pesquisa em *gazetteers*) obter uma alta taxa de precisão, esta metodologia apenas cobre um único domínio, não conseguindo um bom desempenho em aplicações que envolvam textos de outras áreas. O *GermaNER* é um sistema de anotação de texto *open-source* baseado em mecanismos de *machine learning* que pode ser utilizado a partir da linha de comandos, com o objetivo de ser integrado noutro sistema de PLN para anotar e extrair conteúdo.

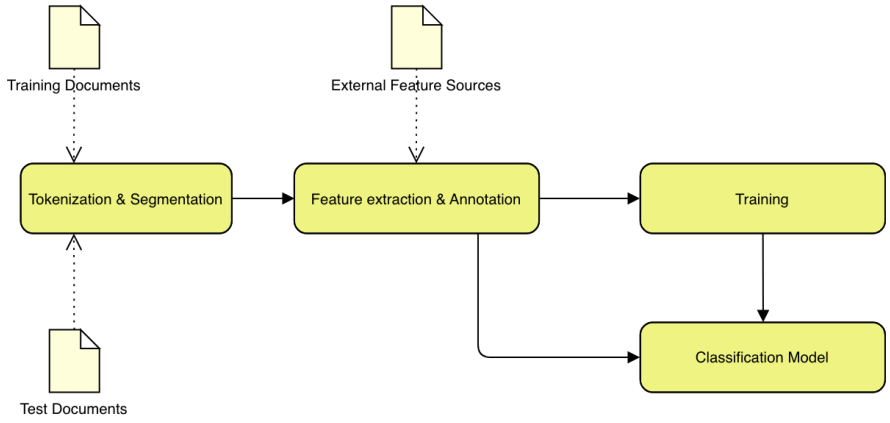


Figura 5: Modelo do sistema de anotação, adaptado de Benikova et al. (2010).

Na Figura 5 podemos ver uma ilustração do sistema de anotação desenvolvido por Benikova et al. (2010). Este sistema foi escrito na linguagem *Java*. Cada um dos seus módulos pode ser executado e configurado individualmente. O primeiro componente do sistema de anotação (*Tokenization & Segmentation*) recebe os documentos de treino e de teste e faz a correspondente separação em palavras e frases. Em seguida, o segundo componente (*Feature extraction & Annotation*) realiza a anotação de entidades através da utilização de dicionários, obtidos a partir de modelos de *machine learning* não supervisionados. Por fim, o último

componente (*Classification Model*) é responsável por aperfeiçoar o sistema de *machine learning* utilizado ao longo do processo de anotação a partir da última iteração.

O último exemplo analisado consistiu no trabalho realizado por Ahmadi and Moradi (2015), um sistema de anotação para textos escritos em língua persa. À semelhança do que acontece com o estudo de PLN em textos de língua portuguesa, os autores encontraram dificuldades em obter estudos semelhantes em língua persa, devido à grande ausência de mecanismos de NER neste idioma. Deste modo, o sistema de anotação de texto persa foi desenvolvido numa abordagem híbrida, que reconhece entidades a partir de um módulo baseado em pesquisas com dicionários de termos e de um módulo de *machine learning* a partir da utilização do modelo oculto de Markov.

O modelo oculto de Markov é um modelo estatístico de geração de uma sequência de dados ocultos a partir de parâmetros observáveis; ou seja, com a utilização deste modelo os autores pretendiam anotar palavras que não estavam presentes nos dados de treino, através de algumas semelhanças com palavras previamente anotadas. A arquitetura seguida por Ahmadi and Moradi (2015) pode ser observada na Figura 6.

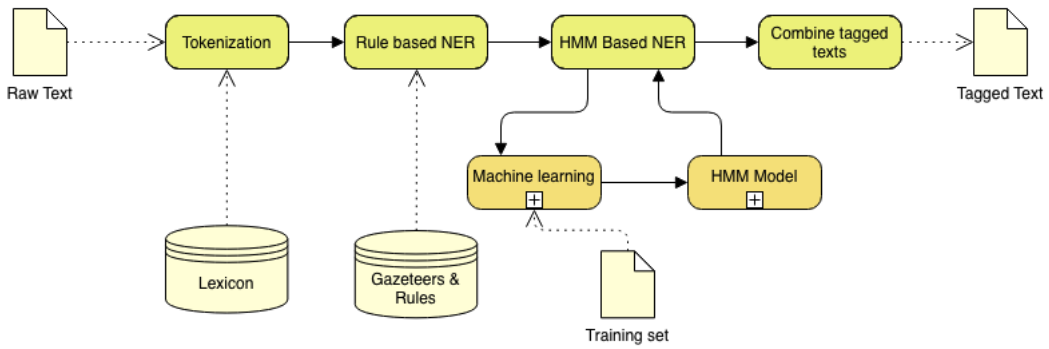


Figura 6: Modelo do sistema de anotação, retirado de Ahmadi and Moradi (2015).

Segundo os autores, a falta de conjuntos de dados anotados em textos persas foi um grande obstáculo no desenvolvimento do modelo de *machine learning* que pretendiam implementar. Assim, para minimizar este problema, foi necessário fazer a criação dos seus próprios conjuntos de dados a partir da anotação manual de documentos. Estes conjuntos de dados serviram como dados de treino e de teste para se utilizarem nas fases posteriores.

Em suma, é possível verificar que os três modelos analisados possuem uma característica comum: foram desenvolvidos de raiz, com o objetivo de anotar e extrair conteúdo de textos escritos em linguagem pouco estudada no campo de PLN. Esta é uma característica partilhada com o sistema de anotação desenvolvido no âmbito da presente dissertação.

3.3 O MODELO DESENVOLVIDO

A partir do estudo realizado na secção anterior, foi possível desenvolver um modelo para o sistema de anotação da aplicação *Tommi* (Barros et al., 2020) com o objetivo de anotar e extrair conceitos dos fólios presentes na sua base de dados documental, referentes aos textos do *Livro das Propriedades* da Mesa Arcebispal de Braga (Barros, 2019), (Barros, 2021).

Na Figura 7 podemos ver um diagrama BPMN relativo ao processo de anotação automática de textos que está integrado na plataforma *Tommi*. A construção deste processo teve como base a modelação do sistema híbrido a partir da utilização de várias técnicas de processamento de texto (Figura 4 da Secção 3.2). O sistema de anotação incorpora várias tarefas. Estas estão divididas em dois componentes distintos, mas interligados entre si, nomeadamente o pré-processamento e o reconhecimento de entidades (NER). O pré-processamento é composto pelas tarefas de separação das palavras dos textos (*Tokenization*) e a sua classificação morfológica.

Para a realização destes trabalhos foram integradas algumas funcionalidades da ferramenta *LinguaKit* (Gamallo and Garcia, 2017). Por outro lado, o NER do sistema incorpora as tarefas de anotação automática (*Automatic Tagging*) e a anotação baseada em regras (*Rule Based Tagging*).

A primeira tarefa do sistema (*Pre-processing*) utiliza dicionários externos (*gazetteer*) para realizar a anotação dos textos com base na procura de palavras nos dicionários disponíveis. A segunda tarefa (NER) utiliza um conjunto de regras pré-definidas que indicam se uma palavra pode ser etiquetada. Por fim, a última tarefa (*Learning*) corresponde à etapa de aprendizagem e ao consequente melhoramento dos *gazetteers* usados ao longo do processo de anotação. Assim sendo, ao longo de sucessivas iterações, o sistema consegue apresentar ao utilizador os fólios que foram anotados através dos seus mecanismos.

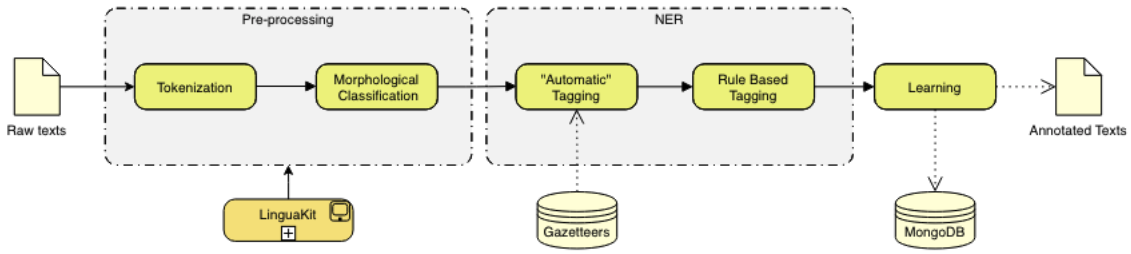


Figura 7: Diagrama BPMN do sistema de anotação de texto.

3.4 ANOTAÇÃO DE ENTIDADES

O processo de anotação de entidades é o componente mais relevante do sistema de anotação de textos. Este componente é composto por duas tarefas de identificação de entidades que são executadas de forma individual e autónoma.

A primeira tarefa consiste num processo de anotação "automático" que recorre a dicionários externos (*gazetteers*) para efetuar a identificação de entidades. No sistema, cada etiqueta ou *tag* encontra-se associada a um *gazetteer*, no qual são armazenadas as palavras que pertencem à classe gramatical dessa *tag* — por exemplo, "**Manuel**"corresponde à *tag* "**nome**", enquanto "**Mirandela**"corresponde à *tag* "**localidade**". Para que o componente de anotação automática possa atuar de forma correta, é necessário realizar o pré-processamento do documento. Este processamento faz a divisão do texto em palavras. Posteriormente, o sistema encarrega-se de, para cada palavra do texto, procurar em todos os *gazetteers* se a palavra está contida em algum dos dicionários. Em caso positivo, a palavra em questão é anotada com a respetiva *tag*.

A segunda tarefa é responsável por realizar um processo de anotação "manual", que recorre a identificadores de classe (e.x. "Santa, "São", "Dona", "Doutor", etc.) que possam estar presentes ao longo do texto. Assim que o sistema deteta um destes identificadores, existe uma elevada probabilidade de que as palavras subsequentes sejam alvo de anotação. À semelhança do que acontece na tarefa anterior, cada *tag* está associada a um conjunto de identificadores de classe, por exemplo, o identificador "**Dona**"pertence à *tag* "**nome**"e o identificador "**Santa**"pertence à *tag* "**localidade**".

Além disso, antes da execução deste componente, deve ser realizado o pré-processamento do texto, no qual é fundamental fazer-se a classificação dos verbos, pronomes, adjetivos, advérbios e determinantes de frase. A partir deste processamento, o sistema assume que as palavras compreendidas entre um identificador de classe e um caso de paragem (um sinal de pontuação ou um verbo) são anotadas com a correspondente *tag*.

Este casal possuem <nome> Domingos Anes </nome> e sua mulher <nome> Isabel Rodrigues </nome>, moradores em o dito lugar de <localidade> São Pedro </localidade>, por título de prazo, o qual não mostraram por dizerem o terem em casa de um <profissão> escrivão </profissão> em <localidade> Chaves </localidade>, mas que eram nele a terceira vida e que pagavam de todas as terras ao <nome> Padre Manuel </nome>

Figura 8: Exemplo de um excerto anotado após a execução dos mecanismos de *tagging*.

Na Figura 8 podemos ver o resultado da aplicação do sistema sobre um excerto retirado do *Livro das Propriedades*. Neste exemplo, a partir do funcionamento do mecanismo de anotação "automática", o sistema foi capaz de identificar os nomes "**Domingos Anes**" e "**Isabel Rodrigues**", a localidade "**Chaves**" e a profissão "**escrivão**", uma vez que eram palavras que constavam nos dicionários das respetivas etiquetas. Por outro lado, o mecanismo de anotação

"manual" conseguiu identificar o nome "**Padre Manuel**" e a localidade "**São Pedro**", visto que ambas as entidades são compostas por identificadores de classe, nomeadamente, "**Padre**" e "**São**".

4

CASO DE ESTUDO

4.1 APRESENTAÇÃO GERAL

Ao longo dos últimos anos foi desenvolvido um sistema de gestão documental para acolher o conteúdo do *Livro das Propriedades* (Barros, 2019), (Barros, 2021), um impressionante manuscrito do século XVII que contém um inventário detalhado das propriedades rústicas e urbanas dos arcebispos de Braga.

Para melhorar a pesquisa e análise de textos e revelar as várias relações entre os seus diversos elementos textuais, foi incorporado no sistema um conjunto de mecanismos orientados para a anotação da base de dados de documentos, permitindo a criação de um conjunto de *tags* relevantes — indexadas, descobertas e estabelecidas — com base em antropónimos, topónimos, graus de parentesco, propriedades e a sua localização, entre outras. Dessa forma, foi possível manter uma base de *tags* como meio de indexação da informação mais relevante contida no referido códice. Além disso, com base na especificação das *tags* criadas, os mecanismos de anotação permitem analisar os documentos que estão contidos no sistema e, por semelhança, sugerir uma estratégia de anotação global para essas *tags*, bem como gerar um mapa de relações de *tags* que pode ser utilizado para descobrir conteúdo semelhante.

A quantidade e a diversidade dos elementos referidos no livro são surpreendentes: todos os nomes e apelidos, povoações, profissões, tipos de terrenos e edificações, entre tantos outros, são muito importantes para o estudo e aprendizagem da geografia, cultura, economia, arquitetura, religião e língua portuguesa do século XVII. A anotação destes elementos evidencia de forma expressiva a sua localização no tempo e no espaço, bem como as suas potenciais relações, facilitando o estudo do *Livro das Propriedades* e proporcionando aos investigadores, linguistas, professores e alunos um valioso instrumento para o alcance e reforço do conhecimento sobre o códice.

4.2 O LIVRO DAS PROPRIEDADES

O *Livro das Propriedades* (Barros, 2019), (Barros, 2021), também conhecido por *Tombo da Mitra*, é um manuscrito que contém informação acerca das propriedades da Mesa Arcebispal de Braga, pertencente ao Arquivo Distrital de Braga e datado do século XVII.

Nesta época, as propriedades da Mesa Arcebispal estendiam-se para além da zona de Braga, abrangendo o Minho e chegando até Trás-os-Montes, atingindo ainda o bispado do Porto, terras de Santarém e até mesmo da Galiza. O registo das propriedades, foros e rendas da Mesa Arcebispal foi feito de uma forma extensa e pormenorizada. Todos os fólios do livro estão rubricados, tendo o códice sido devidamente encerrado e assinado em 1606. Cada fólio (ou seja, em termos leigos cada folha do manuscrito, com a mesma numeração para ambos os lados, expressa apenas no rosto) corresponde a duas páginas de tamanho A3. O manuscrito é composto por 644 fólios, que fazem referência aos tipos de terras (agrícolas, de mato, pasto, etc.), acidentes do terreno e a outros destaques geográficos (penedo, rego, fonte, ermida, quebrada, outeiro, outeirinho, etc.), os nomes de ruas, lugares, rios, povoações, proprietários, os apontamentos biográficos e genealógicos, as indicações dos produtos semeados, os tipos de árvores existentes, a descrição das casas e as suas características (citadinas, rústicas, de morada, de gado, de despejo e celeiro, de adega, sobradadas/terreiras, telhadas, colmadas, etc.). Como tal, a edição do manuscrito (processo que ainda está em curso) é muito importante para o estudo geográfico, sociocultural, agrícola, económico, arquitetónico e religioso de Braga, do Minho e de outros territórios portugueses que a Mesa alcançava (Barros, 2019), (Barros, 2021). Através da edição do manuscrito é possível criar uma base de dados documental, bem como um glossário, rico em termos frequentemente utilizados no século XVII, incluindo longas listas de nomes das terras e respetivos cultivadores, que são importantes para a genealogia das famílias bracarenses, minhotas e transmontanas.

Ainda segundo Barros (2021), o manuscrito conta com 1288 páginas que detalham com grande pormenor e precisão as propriedades rústicas e urbanas das comarcas de Valença, Vila Real, Chaves e Braga. Além disso, também são especificadas as rendas e os pagamentos das respetivas propriedades, devido ao seu emprazamento. Pelo grande detalhe que esta informação encerra, é possível que investigadores ou leitores do *Livro das Propriedades*, caso possuam raízes nestas localidades, consigam encontrar referências ou propriedades dos seus antepassados.

Como referido anteriormente, o *Livro das Propriedades* ou *Tombo da Mitra* encontra-se no Arquivo Distrital de Braga, onde pode ser consultado sob pedido. De forma a perceber a sua exclusividade, singularidade e tamanho, a Figura 9 apresenta o manuscrito e algumas das características enumeradas.

Figura 9: *Livro das Propriedades* fechado.

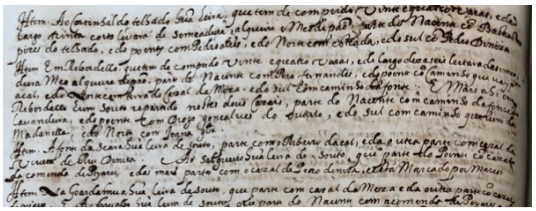
Tal como se pode observar pelas Figuras 9 e 10 (imagens cedidas pela professora Anabela Barros), verificamos que o *Livro das Propriedades* é um códice com uma estrutura curiosa e peso elevados, que contém a minuciosa e extensa relação das propriedades, rendas e foros da Mesa Arcebispal de Braga (Barros, 2019), (Barros, 2021).

Figura 10: *Livro das Propriedades* aberto.

Após a visualização destas imagens, verifica-se que apenas especialistas, com experiência de leitura de manuscritos, conseguem ler e interpretar a informação contida no *Tombo*, o que permite desde logo salientar a dificuldade do leitor comum para pesquisar e recuperar a informação nele contida. Deste modo, o desenvolvimento e disponibilidade de um sistema capaz de armazenar, processar e indexar o texto contido nos documentos do manuscrito é

importante, para que a recuperação da informação seja executada de uma forma simples, eficaz e precisa.

Na Figura 11, à esquerda, apresenta-se um excerto do fólio 97 do *Livro das Propriedades*, escrito em português clássico, e à direita encontra-se a respetiva edição semidiplomática, realizada pela professora Anabela Barros. Através da observação destas imagens, torna-se evidente a dificuldade de interpretação do manuscrito para um leitor comum, mesmo para fólios em bom estado de conservação. Uma vez que o manuscrito data de inícios do século XVII — a escrita utilizada na época era o português clássico — percebemos que a sua leitura e interpretação nos dias de hoje se torna um processo lento e complexo.



Ittem. Ao Cortinhal do telhado hua' Leira, que tem de comprido vinte e quatro varas, e de / largo trinta e oito leuará de semeadura, alqueire e Meo de pão, parte do Nacente co' Bastião / pires do telhado, e do poente com Pedro alres', e do Norte com estrada, e do sul co' Pedro Diniza

Ittem Em Rebordello, que tem de comprido vinte e quatro varas, e de largo dezaseis leuara de semeadura / Meo alqueire de pão, parte do Nacente com Pero fernandes, e do poente co' Caminho que uay p^{a} a / cal, e do Norte com terra do Cazal da Meza. e do sul com caminho da fonte: E mais ahi em / Rebordello hum souto repartido nestes dous Cazais, parte do Nacente com caminho da fonte da / lauandeira, e do poente com Diogo goncalues

Figura 11: Excerto do fólio 97 e a sua transcrição.

Além das dificuldades enumeradas, os estudiosos do *Tombo da Mitra* enfrentam outras barreiras que dificultam a leitura e o manuseamento do manuscrito. O facto do livro não estar reproduzido em formato digital e apenas poder ser consultado presencialmente no Arquivo Distrital de Braga faz com que apenas uma pessoa de cada vez o possa estudar. Por outro lado, esta mesma razão leva a que pessoas distantes da cidade tenham de se deslocar para proceder à consulta do livro, o que por vezes pode levar à desistência do seu estudo. Com o aumento do número de investigadores e devido à importância do códice, surgiu a oportunidade de armazenar o conteúdo do manuscrito num formato digital, derrubando todas as dificuldades para o estudo do *Tombo da Mitra* enumeradas anteriormente. Tendo isso em consideração, idealizou-se e implementou-se o *Tommi* (Barros et al., 2020), uma plataforma *on-line* especialmente orientada para o armazenamento, indexação e pesquisa do conteúdo do *Livro das Propriedades*.

Na próxima secção, será apresentado o sistema *Tommi*, bem como as suas funcionalidades e a forma como foi integrado o sistema de anotação automática de textos, desenvolvida no âmbito da presente dissertação.

4.3 O SISTEMA TOMMI

O sistema *Tommi*[1] (Barros et al., 2020), é uma plataforma *Web* com capacidade para armazenar, indexar e pesquisar os documentos textuais do *Livro das Propriedades* da Mesa Arcebispal de

1 tommi2.di.uminho.pt

Braga. O desenvolvimento deste sistema iniciou-se em 2019, a partir de um projeto interno realizado no curso de Engenharia Informática da Universidade do Minho, coordenado pelo professor Orlando Belo, do Departamento de Informática, e pela professora Anabela Barros, do Departamento de Estudos Portugueses e Lusófonos, da mesma Universidade. Além de coordenadora do projeto, a professora Anabela é a autora da ideia base, uma vez que é a editora do *Livro das Propriedades*, tendo enfrentado pessoalmente as dificuldades enumeradas no capítulo anterior.

O desenvolvimento do projeto teve como requisito a possibilidade de disponibilizar, através de uma ferramenta computacional, meios para que os investigadores do *Livro das Propriedades* pesquisassem e analisassem a informação contida nos seus fólios de uma forma fácil e intuitiva. Além disso, esta ferramenta iria permitir que o conteúdo do manuscrito pudesse ser consultado por utilizadores comuns (não só os investigadores) com interesses históricos, que de outra forma não o poderiam fazer.

Assim, o sistema *Tommi* apresenta-se como uma solução para os problemas detetados, uma vez que oferece mecanismos simples, intuitivos e eficazes para armazenar, recuperar e estudar a informação contida no *Tombo da Mitra*, fornecendo a edição semidiplomática (conservadora) e a edição interpretativa (com grafia atualizada) de vários fólios ou textos em português seiscentista. O sistema *Tommi* pode ser acedido a partir do endereço *tommi2.di.uminho.pt*, utilizando um qualquer navegador *Web* atualmente disponível. A partir do *Tommi* o conteúdo do *Livro das Propriedades* passa a estar acessível a várias pessoas simultaneamente, em qualquer parte do mundo, a qualquer momento.

Através dos trabalhos da presente dissertação serão suprimidas algumas necessidades do sistema, uma vez que, a partir de mecanismos de anotação de texto, os processos de análise, interpretação e recuperação do conteúdo presente no *Tombo da Mitra* ficarão mais facilitados. O sistema de anotação disponibilizará um conjunto de meios capazes de identificar lugares, tipos de terrenos, nomes de pessoas e profissões, referidos ao longo dos fólios. Estes conceitos, catalogados com o auxílio de uma estrutura predefinida, sugerida por especialistas, constituirão uma ferramenta muito útil para a diminuição do tempo de análise e de estudo do manuscrito.

4.3.1 *Características Base do Sistema*

O sistema *Tommi* é capaz de funcionar nos sistemas operativos atuais mais relevantes, nomeadamente o *Microsoft Windows*, *MacOs* e *Linux*. Tal como aconteceu com outras plataformas *Web*, o *Tommi* foi projetado e implementado a partir de uma abordagem cliente-servidor, integrando na parte do servidor (*backend*) todos os mecanismos e estruturas de gestão de dados do sistema, e na parte do cliente (*frontend*) todos os módulos para carregamento e pesquisa de documentos, através de navegadores *Web* convencionais.

A partir da utilização de um navegador *Web* convencional, assim que seja introduzido o endereço *tommi2.di.uminho.pt/login*, será possível aceder à página de autenticação, que dá acesso a todas as funcionalidades do sistema (Figura 12).

Figura 12: Página de autenticação.

Para se utilizar o sistema é necessário ter um conjunto de credenciais válidas (nome de utilizador e senha), definido de acordo com um dos perfis de acesso do sistema. No *Tommi* existem dois perfis distintos:

1. Administrador, que permite visualizar, alterar, editar ou eliminar todas as informações disponíveis no sistema;

2. Leitor, que possui um acesso mais restrito e que pode visualizar (ler) as informações disponíveis, tendo acesso e controlo do seu próprio perfil de utilização.

Para se poder aceder ao sistema, é necessário requisitar previamente ao administrador as senhas de acesso, uma vez que o *Tommi* não disponibiliza um mecanismo de registo aberto, isto é, apenas pessoas autorizadas pelo administrador podem adquirir as credenciais de acesso.

Após a fase de autenticação, o sistema redireciona o utilizador para a página principal interna (Figura 13). No painel central da página, podemos observar um resumo da informação contida no sistema. Esta informação é atualizada sempre que existam alterações nos documentos. Portanto, o utilizador terá sempre acesso ao estado mais recente do sistema.

Nesta página (Figura 13), à esquerda, temos o menu geral do sistema, onde estão disponíveis todas as funcionalidades do *Tommi*. Desta gama de funcionalidades, salientamos as seguintes:

- Importação de Fólios — na qual é realizada a caracterização e a introdução dos textos editados do *Livro das Propriedades* na base documental do sistema;

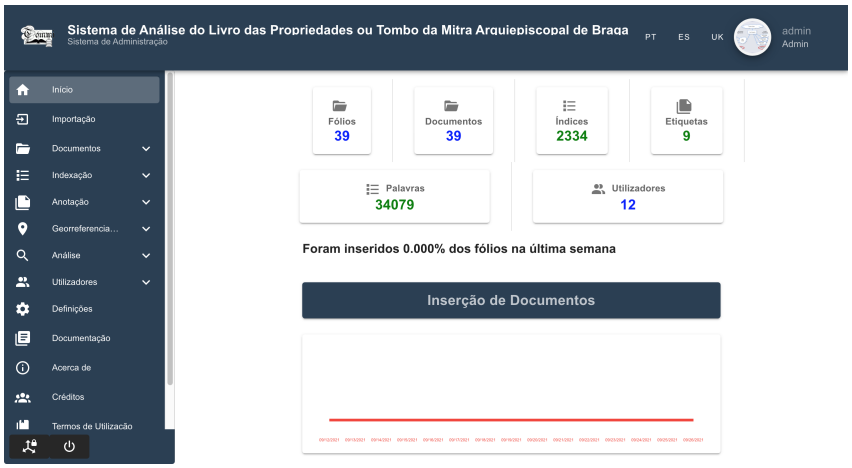


Figura 13: Página principal.

- Gestão de Fólios — em que podemos consultar os documentos presentes na base documental do sistema;
- Gestão de Índices — que fornece mecanismos de pesquisa dos termos localizados nos documentos armazenados;
- Gestão de Utilizadores — que atribui credenciais a novos utilizadores e realiza a gestão das atividades do sistema.

De seguida, apresentaremos de forma mais detalhada cada uma das funcionalidades principais do *Tommi*.

Importação de Fólios

A inserção de documentos no sistema é uma das principais funcionalidades do *Tommi*. Para que esta tarefa possa ser realizada é necessário cumprir um conjunto de passos pré determinado que o sistema irá apresentar ao utilizador. Em primeiro lugar, o utilizador deve assegurar que o texto a ser inserido se encontra num formato aceite pelo sistema, caso contrário a importação falhará logo na primeira etapa. Assim que é feita esta verificação, é possível avançar para o processo de importação do documento. Para o iniciar, o utilizador deve aceder à opção "Importação" disponível no menu principal do *Tommi*. A importação de textos no sistema é realizada em seis passos. O primeiro passo é a catalogação do documento (Figura 14), que, basicamente, consiste em introduzir os metadados do mesmo. Os metadados contêm a informação básica que é necessária para descrever um fólio armazenado no sistema, como, por exemplo: o nome e o número de um fólio, o lado (rosto ou verso) e o tipo do documento (edição semidiplomática ou interpretativa). Opcionalmente, o sistema permite inserir uma foto do fólio, bem como acrescentar algumas notas adicionais sobre o documento a inserir.

Figura 14: Importação de um documento — Passo 1 (Catalogação).

Assim que a primeira etapa é finalizada, segue-se o segundo passo do processo de importação de um documento. Este passo permite fazer uma pré-visualização do documento inserido (Figura 15). Caso tenha sido inserida uma fotografia do documento, a mesma será apresentada posteriormente, no passo 3. Caso contrário, o utilizador será direcionado diretamente para o passo 4, assim que confirmar a informação apresentada pelo sistema.

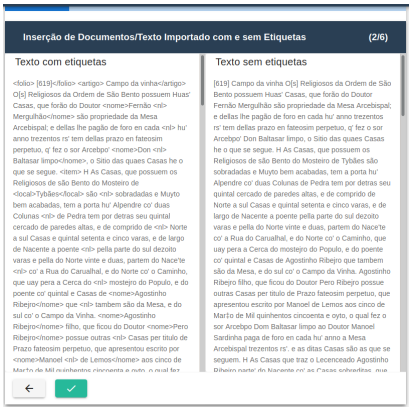


Figura 15: Importação de um documento — Passo 2 (Visualização).

O passo 4 (Figura 16) consiste na identificação de eventuais etiquetas que estejam contidas no documento que pretendemos inserir, mostrando em formato tabular o resultado do processo.

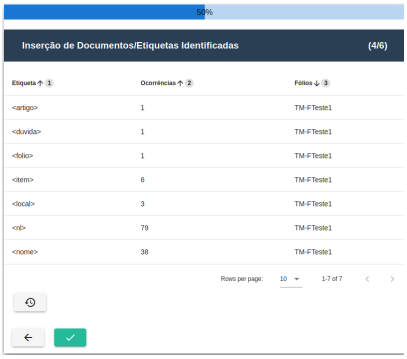


Figura 16: Importação de um documento — Passo 4 (Identificação de etiquetas).

De seguida, no passo 5 (Figura 17), é realizado um processo de identificação semelhante ao realizado no passo anterior, mas agora fazendo a identificação de todas as palavras contidas no documento, indexando-as, para que de futuro sejam facilmente utilizadas pelo mecanismo de pesquisa do *Tommi*.

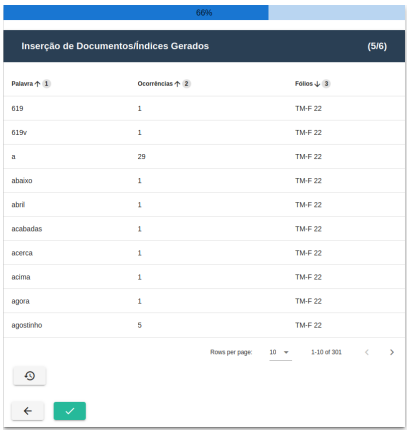


Figura 17: Importação de um documento — Passo 5 (Identificação de palavras).

Após a realização destes passos, o sistema apresenta um resumo da informação importada e guarda de seguida o documento e os seus respetivos metadados na base de dados documental, não existindo possibilidade de um retrocesso para o estado inicial.

Gestão de Fólios

A funcionalidade de gestão de fólios, a que se pode aceder a partir da opção "Gestão de Fólios", na aba "Documentos"do menu principal do sistema, permite ao utilizador visualizar ou remover os documentos armazenados no sistema, bem como as suas fotografias, caso existam. Esta funcionalidade permite ainda a impressão da lista de identificadores dos documentos armazenados (Figura 18).

Figura 18: Gestão dos documentos armazenados no sistema.

Gestão de Índices

Através da opção ”Gestão de Índices”da aba ”Indexação”do menu do sistema (Figura 19), o utilizador pode consultar o catálogo de todos os índices presentes nos documentos inseridos na base de dados documental. Para cada índice ou palavra, esta funcionalidade detalha o número de ocorrências em todo o sistema. Além disso, o sistema é também capaz de identificar a linha e o documento no qual um dado índice foi detetado.

Figura 19: Gestão de índices dos documentos.

Gestão de Utilizadores

A última das funcionalidades principais do sistema *Tommi* é a gestão de utilizadores. Esta funcionalidade pode ser acedida a partir da opção “Gestão de Utilizadores” da aba

"Utilizadores" do menu principal. Após escolher esta opção, o utilizador acede ao ambiente de gestão de dados dos utilizadores (Figura 20). Neste ambiente, tendo-se as credenciais adequadas, podem criar-se novos utilizadores, ver a informação dos utilizadores existentes, editar ou remover um utilizador do sistema.

Figura 20: Gestão de utilizadores do sistema.

4.3.2 *Integração com o Sistema de Anotação Automática de Textos*

Nos últimos dois anos, o sistema *Tommi* evoluiu bastante, tendo sofrido grandes alterações na sua estrutura base e nas suas estruturas de interface com o utilizador. Estas alterações foram provocadas pela adição de novos elementos de dados e de mecanismos capazes de melhorar a análise e a pesquisa dos documentos armazenados no sistema. Tal permitiu que novas ideias de serviços fossem integradas no sistema, nomeadamente as novas áreas de anotação e de georreferenciação de textos.

O sistema de anotação automática de texto integrado no *Tommi* permitirá melhorar a pesquisa e a análise de textos e revelar as várias relações entre conceitos de diferentes *tags*, por exemplo: será possível identificar o nome do proprietário de um terreno, valores de emprazamentos de imóveis à data, bem como outros eventos com interesse histórico. Além disso, com base na especificação das etiquetas (*tags*) criadas, os mecanismos de anotação permitirão fazer a análise de todos os documentos contidos no sistema e, por semelhança, sugerir uma estratégia de anotação global para essas entidades, bem como gerar um mapa de relações de *tags* para descobrir conteúdo semelhante. Desta forma, o utilizador poderá aprofundar o seu conhecimento sobre o *Livro das Propriedades*, tornando-o mais concreto e dinâmico, reduzindo e facilitando as tarefas de interpretação e análise. Como se sabe, estas tarefas são bastante dispendiosas em termos de tempo e dinheiro, se forem realizadas de forma manual por pessoal especializado, como os linguistas.

5

IMPLEMENTAÇÃO DO SISTEMA DE ANOTAÇÃO

O sistema de anotação idealizado e implementado no âmbito desta dissertação, de forma similar ao padrão de desenvolvimento de outros sistemas existentes no mercado, possui quatro módulos distintos, nomeadamente: extração, indexação, classificação e anotação de conteúdo. Para dar início ao processo de desenvolvimento do sistema de anotação, começámos por fazer a anotação das localidades e nomes presentes nos fólios do *Livro das Propriedades*, construindo a base de desenvolvimento do sistema de anotação que pretendíamos desenvolver. Na Figura 21 podemos observar um esquema em Unified Modeling Language (UML) que ilustra as diversas etapas que foram estabelecidas para suportar o processo de anotação dos textos, nomeadamente: seleção, anotação, validação e aprendizagem. Desta forma foi elaborado o diagrama de estados de modo a estabelecer corretamente o comportamento do sistema de anotação de textos. O processo de anotação terá início com a seleção do texto (*Selection*) que o utilizador pretende anotar, para futura análise. De seguida, é realizado o processamento do texto selecionado e a sua subsequente anotação por parte da aplicação (*Tagging*), para que, depois, o utilizador possa validar (*Validation*) a execução do sistema de anotação. Por último, uma vez validada a anotação realizada, os mecanismos de *machine learning* do sistema encarregam-se de atualizar a estrutura interna dos dicionários de termos (*gazetteers*) de forma a reajustar e a enriquecer a base com uma nova anotação (*Learning*).

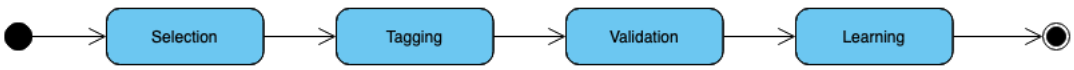


Figura 21: Diagrama do processo de anotação de textos.

De seguida, poderemos observar de forma mais detalhada os diferentes estados do processo de anotação de um texto. Os módulos de seleção (Secção 5.1) e de validação (Secção 5.3) têm um menor grau de complexidade, uma vez que a sua execução corresponde maioritariamente a tarefas de interação com o utilizador. Por outro lado, os módulos de anotação (Secção 5.2) e de aprendizagem (Secção 5.4) têm um elevado grau de complexidade, visto que, durante a sua execução, acontecem os principais processos de anotação e aprendizagem.

5.1 Módulo de Seleção

O módulo de seleção (Figura 22) é o primeiro módulo do processo de anotação de um texto. Este módulo inicia a sua execução com a primeira tarefa (*Text Selection*), que tem como objetivo apresentar ao utilizador os textos disponíveis para anotação. Na segunda tarefa (*Text Selected*), após o utilizador escolher o texto, o sistema carrega em memória a totalidade do mesmo, para o processamento.

A par do módulo de validação, este módulo exige pouca carga computacional, uma vez que a maior parte do trabalho é realizada pelo utilizador, que tem como função escolher e indicar o texto que entrará no processo de anotação. A maior parte do trabalho computacional corresponde ao armazenamento do texto em memória, uma abordagem que foi analisada ao pormenor, já que o carregamento de um ficheiro demasiado grande poderia causar problemas no funcionamento do sistema. No entanto, este acontecimento não criou grandes preocupações, visto que os documentos armazenados no *Tommi* (Barros et al., 2020) não ultrapassam os 300Kb, permitindo que a carga imputada seja apenas residual.

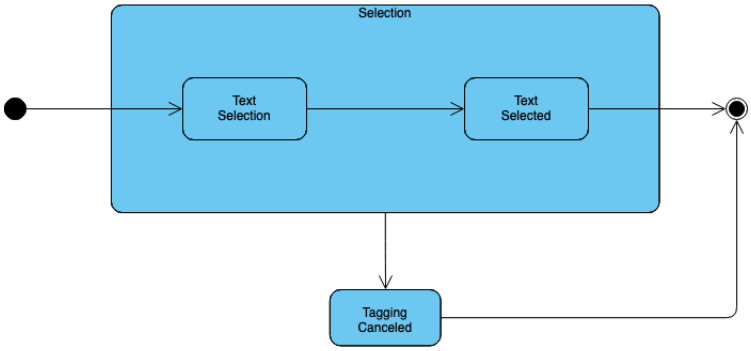


Figura 22: Diagrama do módulo de seleção.

5.2 Módulo de Anotação

O processo de anotação (Figura 23) começa com a classificação do texto (*Text Classified*). Nesta primeira etapa o sistema divide as palavras e agrega-as numa estrutura de dados específica, para que possam ser utilizadas nas outras tarefas de anotação. As tarefas de divisão e de agregação das palavras foram implementadas com a ferramenta de processamento de linguagem natural *LinguaKit* (Gamallo and Garcia, 2017). De seguida, numa segunda etapa (*Text Modernized*), o sistema moderniza as palavras do texto, que estão escritas em português clássico, de forma a permitir a utilização de dicionários mais recentes, redigidos em português atual. Ao atingir a tarefa "*Text Annotated by Dictionaries*", o sistema realiza a anotação das palavras com base nas informações contidas nos dicionários, que se encontram

armazenados na base do sistema (Anotação Automática). Cada uma das *tags* que foram criadas está associada a um dicionário específico, cujas palavras se enquadram no contexto de aplicação da *tag*. Ao longo desta parte do processo, o sistema realiza esta tarefa sobre cada palavra que está presente nos dicionários, fazendo uma comparação direta com as palavras que estão no texto em processamento. Esta tarefa verifica se estas últimas estão presentes no dicionário. Caso as palavras se encontrem no mesmo, são então anotadas com a *tag* que pertence ao dicionário em causa.

Em seguida, a partir de regras de anotação manual (Secção 5.2.4) que definem a forma como cada elemento pode ser anotado, o sistema inicia uma nova tarefa (*Text Annotated by NER*), fazendo uma segunda anotação para identificar os elementos que, por algum motivo, não foram associados a uma *tag*. Por fim, na última tarefa da anotação (*Text Agregation*), o sistema faz a agregação das palavras anotadas que estão na memória de trabalho da aplicação, de forma a incluir todas as *tags* definidas durante o processo de anotação do texto original.

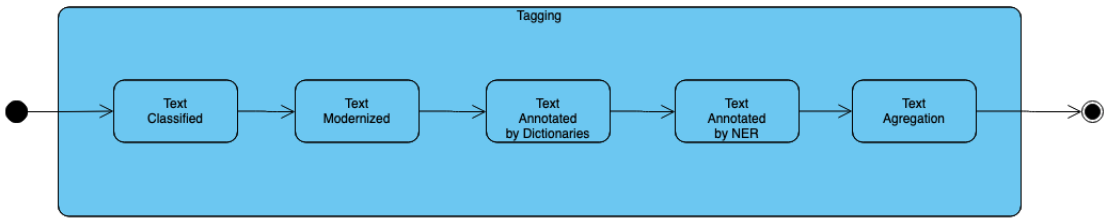


Figura 23: Diagrama do módulo de anotação.

5.2.1 *Classificação do Texto*

Nesta tarefa o sistema começa por fazer a separação do texto em palavras, com o objetivo de criar uma estrutura de dados adequada a todo o processo de anotação e que possa ser utilizada nos próximos estados, como, por exemplo, na anotação automática (Secção 5.2.3) e na anotação manual (Secção 5.2.4).

Esta tarefa utiliza a ferramenta *LinguaKit* (Gamallo and Garcia, 2017), porque apresenta as funções necessárias para dividir o texto em palavras e para fazer a sua classificação morfológica. A classificação morfológica de palavras indica se uma palavra é um nome, um adjetivo, um verbo, um advérbio, etc.

Na estrutura de dados que foi criada, para além da classificação das palavras, também se armazena a posição de cada uma das mesmas, bem como a sua forma homogeneizada (em minúsculas e sem acentos). A forma homogeneizada será utilizada ao longo do processo de pesquisa nos dicionários, juntamente com um valor lógico (verdadeiro ou falso), que informa se essa palavra já foi anotada ou não. Inicialmente, este valor assume o valor de falso, tendo um papel fundamental no funcionamento do sistema de anotação. Assim que o

módulo de anotação automática termina a sua execução, e começa a execução do módulo de anotação manual (Secção 5.2.4), este último módulo deve ter o conhecimento sobre quais as palavras que já foram anotadas até ao momento.

5.2.2 *Atualização de Grafia*

Como referimos, os textos do *Livro das Propriedades* foram redigidos em português clássico e os dicionários de localidades que utilizámos estão escritos em português contemporâneo. Tais circunstâncias implicaram fazer a atualização da grafia dos textos, de forma que fosse possível utilizar os dicionários referidos. Como a maior parte das palavras não seguem as normas ortográficas atuais, a deteção de entidades é de difícil execução para o sistema de anotação implementado. Como tal, foi necessário criar no sistema um processo que fosse capaz de executar a modernização das palavras dos documentos de uma forma automática. Devido a isso, este módulo tornou-se num componente essencial do sistema.

Para desenvolver o módulo de *Atualização da Grafia* estabeleceu-se um conjunto de regras de atualização lexical, no qual cada padrão clássico tem a correspondente tradução modernizada. Este processo aplica as regras de atualização de grafia (Tabela 3), guardando como resultado os textos atualizados e uma estrutura de dados (dicionário de conversão), para que no futuro se possa reverter a conversão realizada e, consequentemente, recuperar o formato original.

Clássico	Modernizado
y	i
ll	l
uu	uv
th	t
j [consoante]	l
[vogal] u [vogal]	v
d’	de
co’	com
q’	que
hu’	um
nn	n
ee	e

Tabela 3: Regras de atualização de grafia.

Todos os textos atualizados possuem um dicionário de conversão, que é constituído por triplos de palavras (clássica, modernizada, posição), para que seja possível aceder à atualização exata de cada palavra, consoante a sua posição. Na Tabela 4 podemos ver um

excerto de um pequeno texto do *Livro das Propriedades* que foi atualizado. Neste caso, as palavras "largo", "banda" ou "sul" não sofreram qualquer alteração, uma vez que nenhuma das regras de atualização pôde ser aplicada a estes exemplos. Porém, a palavra "pella" foi substituída (atualizada) pela palavra "pela", após ter sido aplicada a segunda regra apresentada na Tabela 3.

e	de	largo	pella	banda	do	sul	sessenta	e	quatro
e	de	largo	pela	banda	do	sul	sessenta	e	quatro

Tabela 4: Excerto do *Livro das Propriedades* modernizado.

Finalizada a atualização de grafia, o sistema de anotação avança para o estado seguinte, no qual se procede à anotação automática do texto ou à anotação do texto por meio de dicionários.

5.2.3 *Anotação Automática*

A anotação automática de textos (ou anotação por meio de dicionários) é a segunda etapa do processo de anotação, tendo como objetivo fazer a anotação de entidades, a partir dos dicionários de palavras referentes a cada uma das etiquetas (*tags*) do sistema.

Para que fosse possível realizar esta tarefa teve que se criar um dicionário específico, um *gazetteer*, com todas as localidades do território português, com particular ênfase no desenvolvimento inicial da anotação de localidades, para que, no futuro, a mesma estratégia fosse aplicada às restantes *tags*. Este dicionário foi construído a partir de um *dataset* específico (Portal de Dados Abertos da Administração Pública, 2020), que contém dados nacionais relativos aos distritos, concelhos e freguesias de Portugal. O *dataset* (Tabela 5) que foi processado estava no formato Microsoft Excel Spreadsheet File (XLS) e ocupava cerca de 600Kb de espaço.

Distrito	Concelho	Freguesia
Braga	Esposende	Antas
Braga	Esposende	Forjães
Braga	Esposende	Gemeses
Braga	Esposende	Vila Chã
Braga	Esposende	União das freguesias de Apúlia e Fão
Braga	Esposende	União das freguesias de Belinho e Mar
Braga	Esposende	União das freguesias de Esposende, Marinhas e Gandra
Braga	Esposende	União das freguesias de Fonte Boa e Rio Tinto
Braga	Esposende	União das freguesias de Palmeira de Faro e Curvos

Tabela 5: Excerto do *dataset* referente ao concelho de Esposende.

O processamento do ficheiro foi realizado utilizando um *script* escrito em *Python* (Python, 2021). Após a leitura do ficheiro, o *script* encarrega-se de processar e tratar os dados que neles estão contidos para se criar uma estrutura de dados específica capaz de armazenar o nome de todas as localidades do território nacional, num ficheiro de dados JavaScript Object Notation (JSON). Durante este processo, é criado um segundo ficheiro, também em formato JSON, contendo os nomes das localidades filtradas, isto é, em letra minúscula e com todos os acentos removidos. A homogeneização dos nomes dos locais é importante, visto que aumenta a precisão de acerto na sua anotação, dado que podem surgir localidades com acentos em falta ou escritas somente em minúsculas.

Terminado o processo de criação do dicionário de localidades, passámos ao desenvolvimento do módulo de anotação automática. Para as restantes *tags* do sistema, o apoio da Professora Anabela Barros foi fundamental, disponibilizando listas de termos (Barros, 2021), provenientes de estudos anteriores, que serviram como base para a criação dos dicionários. Deste modo foi possível aumentar a rapidez, a eficácia e a precisão na anotação de nomes, terrenos, profissões, etc.

Após esta tarefa, realizou-se a combinação de palavras, que, basicamente, consiste na junção de palavras seguidas, até um limite pré-definido. Esta tarefa tem que ser realizada, uma vez que existem nomes de localidades e de pessoas que são constituídos por mais do que uma palavra. As localidades ou os nomes portugueses não apresentam, usualmente, mais do que cinco palavras consecutivas. Assim, definimos um máximo de cinco palavras por combinação.

Por fim, para terminar o processo de anotação automática, fez-se a comparação das palavras contidas no texto com os termos registados nos dicionários de *tags*. Como já referido, o processo de comparação utilizou palavras em minúsculas e sem acentos. Com esta estratégia, foi possível cobrir um maior número de casos, aumentando o nível de precisão e de eficácia da anotação.

5.2.4 *Anotação Manual*

Tendo terminado o mecanismo de anotação automática de entidades, o sistema avança para um novo módulo, a anotação manual, de forma a realizar uma segunda ronda de anotação, com vista à identificação de potenciais elementos que não estão presentes nos dicionários e que não foram associados a uma *tag*.

O *Livro das Propriedades* é um manuscrito do início do século XVII, deste modo, ao longo dos seus textos, é muito provável encontrarem-se localidades, nomes, terrenos ou profissões cujas designações mudaram com o passar dos anos. O português utilizado nos textos é seiscentista, portanto, contém termos que atualmente não são utilizados, o que pode fazer com que estes casos não sejam anotados de forma automática. Por exemplo, existem localidades presentes no manuscrito cujos nomes sofreram diversas alterações, ou simplesmente cuja ortografia se revela com variação. Como tal, estas localidades não estão presentes no dicionário de localidades. Para tratar esses casos tivemos que implementar um mecanismo que fosse capaz de detetar essas situações e anotá-las devidamente manualmente.

Na criação do módulo de anotação manual foi implementado um mecanismo de verificação dos indicadores de início e de fim de classe. Os indicadores de início de classe são palavras presentes no texto que antecedem ou pertencem ao termo que se pretende identificar. Por exemplo, durante o processamento das palavras do texto, logo que o sistema deteta um indicador de início de classe, é bastante provável que as próximas palavras possam ser categorizadas por uma dada *tag*. Na Tabela 6 podemos observar alguns exemplos de identificadores de início de classe para as *tags* "localidade"e "nome".

Localidade	**Nome**
Couto	Doutor
Santa	Licenciado
São	Dona
Aldea	Padre
Vila	Arcebispo
...	...

Tabela 6: Exemplos de identificadores de início de classe.

Uma vez detetado o ponto de partida para uma possível anotação, o sistema tem de saber quais são os pontos de paragem. O ponto de paragem é detetado assim que o sistema encontre, por exemplo, um determinante, um verbo ou um adjetivo. Caso o sistema detete uma preposição, é verificada a próxima palavra, a qual, se por sua vez pertencer à classe gramatical dos casos de paragem (Tabela 7), conduz à paragem da anotação desse caso. Na Tabela 7 podemos ver todos os casos de paragem que foram identificados para o mecanismo de anotação manual.

Casos de Paragem	
Simples	Especial
Verbos	Preposições
Nomes Comuns	
Determinantes	
Adjetivos	
Advérbios	
Conjunções	
Pontuação	

Tabela 7: Exemplos de casos de paragem.

Este mecanismo de anotação manual tem um funcionamento muito semelhante ao mecanismo de anotação automática apresentado anteriormente. Porém, em vez de ler os dicionários, o mecanismo de anotação manual inicia o seu processo com a leitura dos identificadores de início de classe e dos casos de paragem, armazenando-os em estruturas de dados adequadas. De seguida, o módulo reutiliza a classificação do mecanismo anterior, onde foram organizadas as palavras do texto consoante a sua classe morfológica, o que lhe permite encontrar os casos de paragem contidos nos textos e, desta forma, detetar as potenciais entidades para anotar. No fim deste processo obtemos um texto anotado, no qual podemos encontrar anotações como a seguinte "... e do nascente e sul com <**nome**>**Dona Maria** </**nome**> de semeadura". Neste exemplo, o nome anotado foi identificado com base nos identificadores de início de classe, que são as palavras que marcam o início da pesquisa da entidade, neste caso "Dona". Por sua vez, o caso de paragem é a preposição "de", que marca o fim da pesquisa. Após a identificação dos limites referidos, o sistema foi capaz de identificar o nome "Dona Maria"e anotá-lo corretamente com a respetiva *tag*.

5.2.5 *Agregação do Texto*

Terminadas as duas fases de anotação (Automática e Manual), o mecanismo de etiquetação (*Tagging*) avança para o último estado, a agregação do texto.

Esta tarefa consiste em fazer a agregação das palavras anotadas e não anotadas, que estão guardadas na memória do sistema, numa estrutura de dados semelhante àquela que se apresenta na Figura 24. Como se pode observar, a estrutura de dados possui um conjunto de atributos para cada uma das palavras. Desse conjunto, destacamos os atributos *"word"*, *"s_tag"*, *"position"*, *"annotated"*, *"tipo"*e *"color"*. Os primeiros dois atributos representam a forma da palavra, que pode conter ou não uma *tag*. Por exemplo, para uma palavra não anotada, ambos os atributos irão apresentar o mesmo valor. No entanto, o mesmo não se verifica no caso de uma palavra anotada, sendo que o primeiro atributo irá conter a palavra "rodeada"pela *tag*. O atributo *"position"*refere-se à posição da palavra no texto. É um dos principais atributos no processo de agregação do texto, uma vez que é necessária a posição de cada uma das palavras na recuperação do texto original em qualquer processo de anotação. Os três últimos atributos indicam, respetivamente, se uma palavra está ou não anotada (*annotated*), o tipo de *tag* e a cor que irá ser exibida na interface do sistema de anotação.

```
{
    "word": "<nome> Marcos </nome>",
    "s_tag" : "Marcos",
    "position": 346,
    "anotated": true,
    "tipo": "nome",
    "color": "green",
    "selected": false
}
```

```
{
    "word" : "depois",
    "s_tag" : "depois",
    "position" : 332,
    "anotated" : false,
    "tipo" : "",
    "color" : "black",
    "selected" : false
}
```

Figura 24: Exemplo de uma estrutura de dados de uma palavra anotada e de uma palavra não anotada.

Na Figura 25 podemos ver o resultado da execução da tarefa de agregação de texto. Caso seja necessário, neste ambiente podemos fazer a recuperação do texto original que foi anotado.

Figura 25: Exemplo de um texto anotado no sistema.

5.3 MÓDULO DE VALIDAÇÃO

O módulo de validação tem como objetivo realizar a verificação da anotação realizada no módulo anterior. Como podemos ver através da Figura 26, este módulo pode seguir três caminhos diferentes, todos eles tendo um denominador comum: o utilizador. O resultado do módulo de validação é unicamente determinado pelo utilizador e, tal como no módulo de seleção, este é outro módulo que exige pouca capacidade computacional do sistema.

Como resultado da escolha do utilizador, o fluxo pode progredir para um estado de cancelamento (*Tagging canceled*), no qual o processo de anotação do texto é interrompido, voltando ao estado inicial antes da anotação. Por outro lado, o utilizador pode aceitar todas as anotações realizadas automaticamente pelo sistema, avançando para o último estado, *Text Validated*. A terceira opção consiste em adicionar anotações manuais ou alterar a anotação de certas entidades. Esta opção (*Text tagged with manual annotation*) acontece quando o utilizador decide realizar alterações manuais, com o intuito de corrigir algumas anotações efetuadas pelo sistema, ou com o intuito de adicionar anotações que, por algum motivo, não foram detetadas.

Por fim, após a validação da anotação, com ou sem alterações manuais, o sistema avança para o último estado: a aprendizagem.

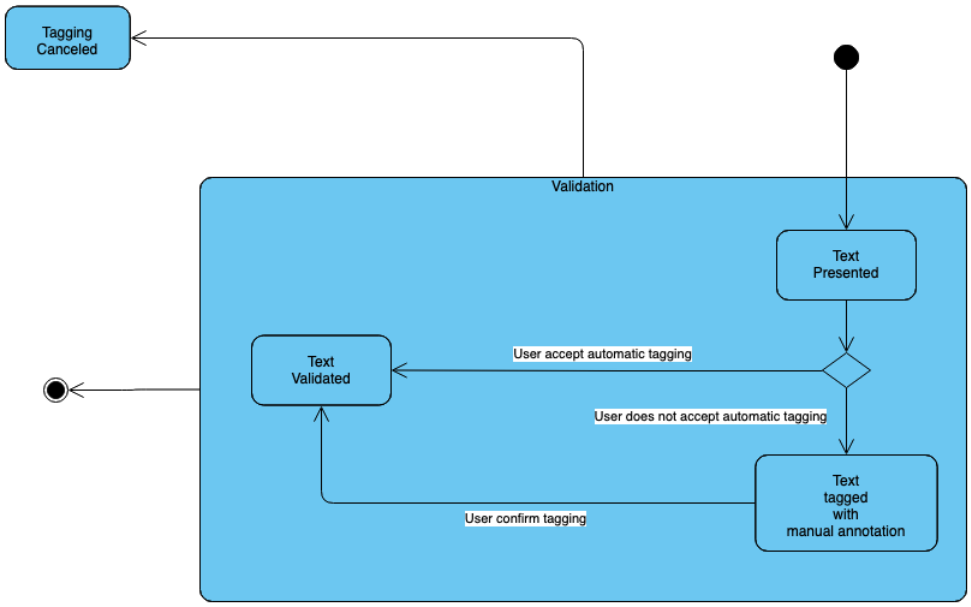


Figura 26: Exemplo do módulo de validação.

5.4 Módulo de Aprendizagem

O módulo de aprendizagem é o último módulo que integra o processo de anotação de textos. O processo de aprendizagem (Figura 27) envolve o armazenamento do texto anotado, durante a atualização do índice de *tags*, e, por fim, a atualização dos dicionários.

Na primeira tarefa (*Text Stored*) é realizado o armazenamento do texto anotado na base de dados documental do sistema. Desta forma, qualquer utilizador pode visualizar o texto anotado sem precisar de executar uma tarefa de anotação em cada acesso. Na segunda (*Tags Index Updated*) é realizada a atualização dos índices de *tags* do sistema, permitindo identificar as ocorrências de uma dada *tag* no sistema. Basicamente, indica os textos em que podemos encontrar localidades, profissões, terrenos, etc. Na última tarefa (*Dictionaries Updated*) faz-se a atualização dos dicionários de *tags* que suportam a tarefa de anotação automática (Secção 5.2.3).

Com a constante atualização dos dicionários, podemos afirmar que o sistema de anotação tem capacidades de aprendizagem. Durante o processo de anotação podem ser descobertas novas entidades, a partir da tarefa de anotação manual (Secção 5.2.4), ou a partir de anotações realizadas pelo utilizador no estado de validação (Secção 5.3). Dessa forma, a descoberta de novas entidades enriquece os dicionários e aperfeiçoa o processo de anotação em futuras iterações.

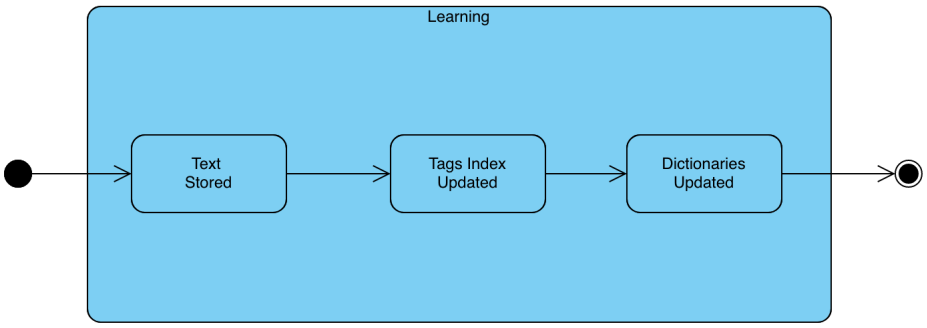


Figura 27: Diagrama do módulo de aprendizagem.

5.5 Integração do Sistema de Anotação

5.5.1 *O servidor de backend*

O servidor de *backend* do sistema *Tommi* foi desenvolvido a partir da *framework Flask* (Pallets, 2021), uma ferramenta orientada para o desenvolvimento de aplicações *web*, suportada pela linguagem *Python* (Python, 2021). Dado que o servidor de *backend* do *Tommi* foi desenvolvido

em *Python*, a integração do sistema de anotação realizou-se de uma forma bastante simples e direta, já que não foi necessário fazer alterações no código fonte de ambos os sistemas.

Após termos executado a integração do sistema, verificámos que teríamos de implementar novas rotas no sistema *Tommi*, que eram fundamentais para fazer a ligação entre os módulos de anotação, a base de dados documental e a interface do sistema. As rotas que foram adicionadas ao servidor de *backend* do *Tommi* abrangem diferentes tarefas do sistema de anotação de textos. Entre elas, podemos destacar as seguintes:

- Gestão das regras de atualização, que têm como objetivo adicionar, remover ou listar as regras de atualização de grafia;
- Gestão das *tags* do sistema, que realizam o processamento relacionado com as mesmas, como, por exemplo, a adição, edição ou remoção de *tags*, ou a listagem de uma ou mais *tags*.
- Gestão dos fólios anotados e atualizados, que são as rotas responsáveis por realizar a anotação ou modernização de um dado texto.
- Re-anotação da base, que é uma das rotas mais importantes do sistema de anotação. Com a constante atualização dos dicionários de *tags* e consequente melhoramento do sistema de anotação, esta rota permite manter uma base de textos anotados constantemente atualizada.

5.5.2 *O servidor de frontend*

Assim que foram terminadas as alterações no servidor de *backend* do *Tommi*, passou-se à realização de algumas alterações no servidor de *frontend*, com o objetivo de integrar os componentes da interface do sistema de anotação automática de textos. O servidor de *frontend* do *Tommi* foi desenvolvido em *VueJs* (Vue, 2021), uma *framework* de *JavaScript* direcionada para o desenvolvimento de interfaces de aplicações *Web*.

No desenvolvimento de aplicações em *VueJs*, existe uma divisão clara entre o código *HTML* e o código *JavaScript*. Na primeira parte do código, o *HTML* tem como função incorporar os componentes que irão ser apresentados ao utilizador na interface do sistema. Na segunda parte do código, o *JavaScript* estabelece a ponte entre o *frontend* e o *backend*, já que tem a função de realizar os pedidos *REST*[1] ao servidor de *backend*, com o intuito de fornecer os dados necessários aos componentes da interface.

Esta estruturação do código do *frontend* foi realizada nas *views* e componentes do *VueJs*. Os componentes são pequenas peças, como uma tabela ou conjunto de botões, que podem ser personalizadas consoante a necessidade da aplicação e que podem ser reutilizadas em

1 https://restfulapi.net/

diferentes *views*. As *views* têm como função integrar os diferentes componentes, necessários para criar uma página *Web* completa. Além destas duas peças fundamentais, o servidor de *frontend* integra uma terceira peça, que tem, igualmente, um papel fundamental: o *Router*. Esta última peça está encarregue de processar o Uniform Resource Locator (URL) introduzido pelo utilizador no *browser*, de forma que o servidor possa fornecer a página *Web* solicitada.

Para integrar corretamente os módulos da interface do sistema de anotação no servidor de *frontend* do *Tommi*, foi desenvolvido um conjunto de *views* e de componentes para suportar as necessidades do utilizador em processos de anotação de textos. Além disso, foram adicionadas outras rotas ao *Router* do servidor, para que se pudesse definir o URL necessário para a invocação de cada um dos módulos.

As *views* desenvolvidas para a integração do sistema de anotação suportam os diversos módulos do servidor de *backend* que foram criados para o mesmo efeito, ou seja, têm como objetivo atender às necessidades da interface relativamente às funcionalidades de:

- Gestão das regras de atualização — Ambiente responsável por visualizar, adicionar ou remover regras de atualização.
- Gestão das *tags* do sistema — Ambiente responsável por visualizar, adicionar, editar ou remover as *tags* do sistema.
- Anotação e/ou modernização dos fólios — Ambiente onde o utilizador pode executar as tarefas de anotação ou atualização ortográfica de textos.
- Re-anotação da base — Ambiente responsável por re-executar a anotação de todos os textos do sistema.

Para utilizar as funcionalidades do sistema de anotação, o utilizador deve aceder à barra de navegação do sistema, em particular, ao menu de funcionalidades de anotação. O menu do sistema de anotação contém uma entrada para cada uma das funcionalidades descritas anteriormente. A primeira opção desse menu (Figura 28) permite ao utilizador realizar a gestão das regras de atualização do sistema, ou seja, visualizar as regras que estão ativas e que podem ser utilizadas, bem como adicionar novas regras ou apagar outras que deixem de ter utilidade.

Figura 28: Ambiente de gestão de regras de atualização.

A segunda opção do menu de anotação permite apresentar as páginas relativas à gestão das *tags* do sistema. Nesta opção, o utilizador pode visualizar o conjunto de *tags* do sistema, ou adicionar e remover etiquetas. Uma vez que foi desenvolvido um módulo para a criação do dicionário de localidades (Secção 5.2.3), a *tag* "localidade"exigiu um processamento específico em relação às restantes etiquetas. Por isso, o sistema não permite a remoção desta *tag* (Figura 29).

Figura 29: Ambiente de gestão das *tags* do sistema.

Na Figura 30 pode-se ver a caracterização de uma *tag*, neste caso a *tag* "terreno". Neste ambiente, podem fazer-se, também, alterações à configuração da *tag*, como por exemplo,

definir a sua usabilidade no processo de anotação de textos, a partir da mudança do estado (ativa ou inativa), alterar a cor da etiqueta, modificar o dicionário de palavras ou os indicadores de classe. A alteração do dicionário de palavras ou dos indicadores de classe envolve a adição de palavras que o utilizador considere úteis para o sistema no processo de anotação automática e manual.

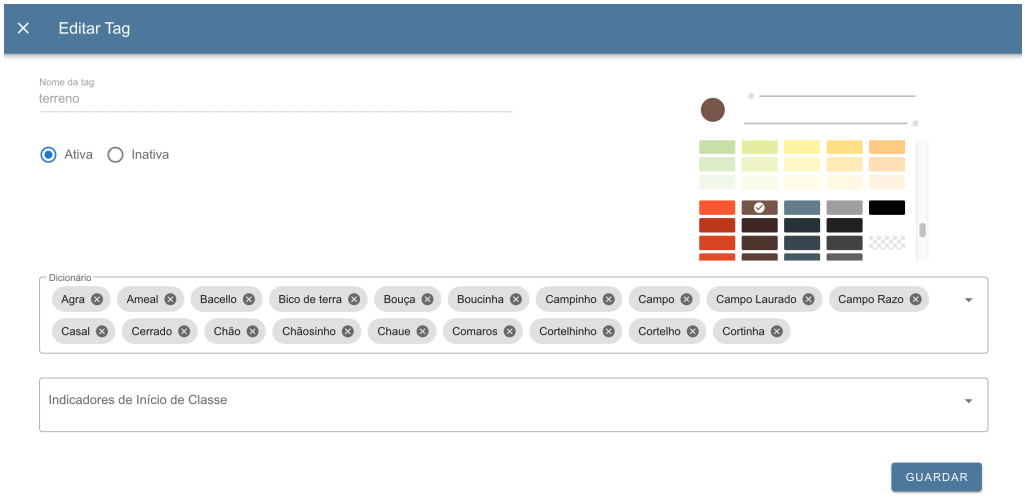


Figura 30: Ambiente de visualização dos detalhes da *tag* "terreno".

A terceira opção dá-nos acesso à anotação e à modernização dos textos. Assim que acedemos à opção "Anotação de fólios", o sistema apresenta-nos um novo ambiente (Figura 31), no qual podemos ver a lista de fólios disponíveis para anotar ou atualizar. Caso o fólio tenha sido anotado e guardado previamente, o sistema disponibiliza uma opção para visualizar o fólio anotado, sem necessitar de executar novamente o processo de anotação, otimizando, assim, o tempo de resposta e diminuindo o processamento dos textos. Esta operação pode ser realizada desta maneira, porque após a validação da anotação do texto por parte do utilizador, o sistema guarda o texto anotado na íntegra na base do sistema (Secção 5.4).

Figura 31: Ambiente do sistema de anotação.

Para melhor compreensão do funcionamento do sistema, vejamos a Figura 32. Esta revela-nos o ambiente principal do sistema de anotação de texto. Assim que um utilizador pede ao sistema a anotação de um determinado texto, este deve aguardar pelo processamento do texto no servidor de *backend*, para que, em seguida, seja redirecionado para o ambiente referido anteriormente. Assim que é redirecionado para este ambiente, o utilizador pode verificar quais as entidades que foram anotadas, podendo alterar as suas etiquetas, se assim o desejar. No fim da verificação, o utilizador, ao carregar no botão "submeter", valida a anotação do texto, e o sistema prossegue para o estado seguinte (Secção 5.4). Caso não seja realizado este passo, a anotação, bem como todas as alterações manuais, serão ignoradas pela aplicação.

Figura 32: Ambiente principal do sistema de anotação.

Na Figura 32 podemos ver um exemplo de uma palavra anotada: "Campinho". Esta palavra foi anotada pelo sistema como sendo uma localidade. No entanto, o utilizador pode fazer a alteração da etiqueta através do editor de *tags* (Figura 33). Para aceder ao editor basta *clicar* numa palavra.

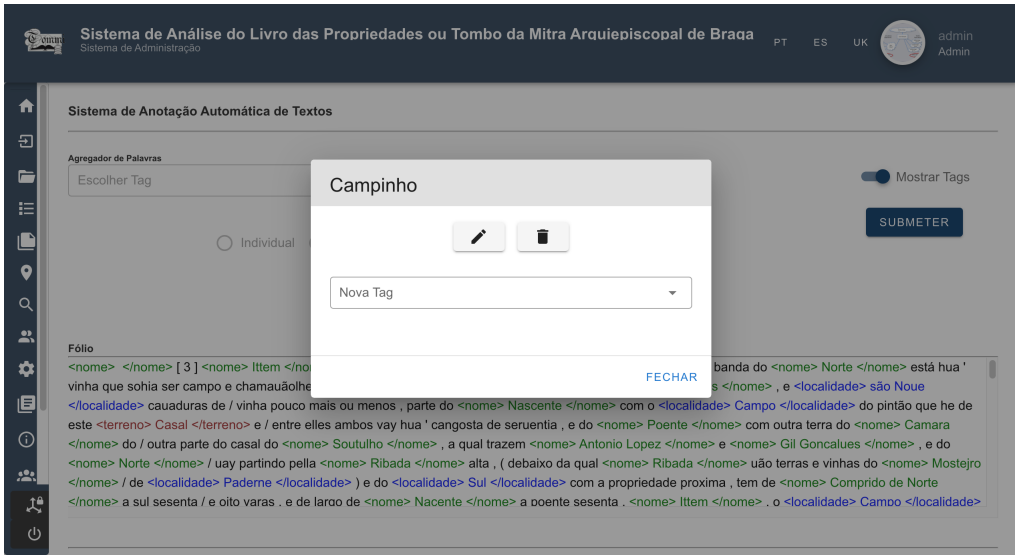


Figura 33: O ambiente do editor de *tags*.

.

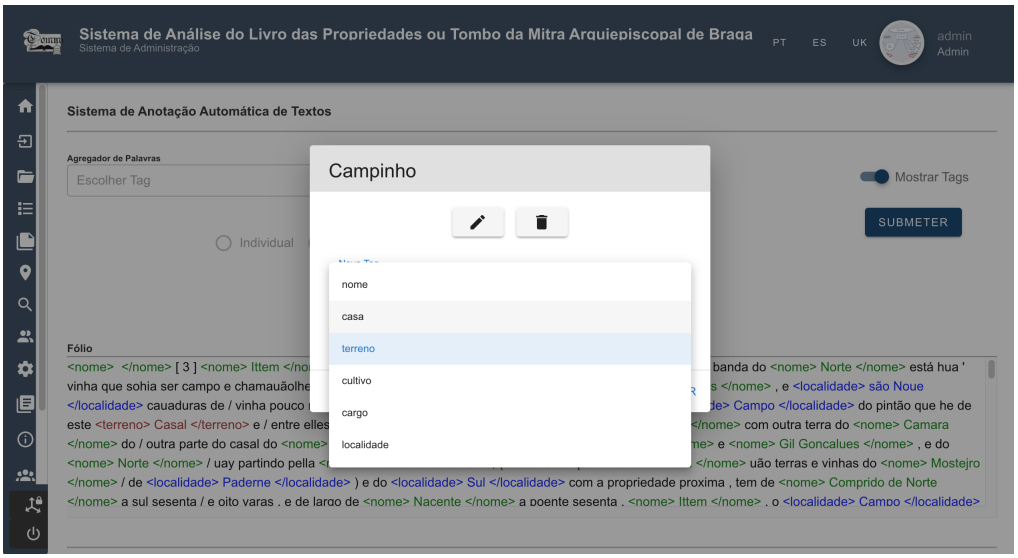


Figura 34: Alteração da etiqueta de uma palavra.

Figura 35: Aspeto do ambiente principal do sistema de anotação após a alteração de uma *tag*.

O resultado da alteração da etiqueta da palavra "Campinho"(Figura 34) pode ser observado na Figura 35. Assim sendo, considera-se que o utilizador realizou alterações manuais. Esta situação afetará a anotação automática em futuros processamentos. No momento em que se efetue a submissão do texto, a palavra "Campinho"será adicionada ao dicionário da *tag* "terreno".

Além das funcionalidades já descritas, o sistema de anotação de texto permite fazer a ocultação das *tags* (Figuras 36 e 37). Quando isso acontece as palavras aparecem coloridas, de acordo com o tipo de *tag* atribuída. Esta funcionalidade permite ao utilizador fazer a validação dos textos anotados de uma forma simples e fluída, o que não acontece usualmente

em textos anotados, dada a presença de elementos externos. Caso as *tags* estejam escondidas, o utilizador pode aceder à *tag* da palavra, colocando o apontador de rato por cima da palavra pretendida (Figura 37).

Figura 36: Sistema de anotação a mostrar as *tags*.

Figura 37: Sistema de anotação a ocultar as *tags*.

A última funcionalidade do sistema, que está presente no seu menu principal, é a re-anotação da base. Com esta funcionalidade podemos executar a anotação de todos os textos do sistema, incluindo aqueles que já tinham sido anotados previamente, com o auxílio dos dicionários mais recentes da aplicação. Visto que o sistema tem a capacidade de melhorar e de aprender com a quantidade de anotações realizadas, o desenvolvimento

desta funcionalidade, permite ao sistema manter uma base de anotação atualizada e eficaz no momento que se executa esta opção.

6

CONCLUSÕES E TRABALHO FUTURO

6.1 CONCLUSÕES

Neste trabalho de dissertação foi concebido e implementado um sistema de anotação automática de textos para o sistema *Tommi* (Barros et al., 2020), cujo principal objetivo é identificar e anotar conceitos referidos nos documentos do *Livro das Propriedades* (Barros, 2019), (Barros, 2021). O sistema desenvolvido possibilita a identificação de localidades, nomes, profissões, terrenos, bem como outras etiquetas com elevado interesse histórico, facilitando o estudo dos textos que contêm o inventário das propriedades da Mesa Arcebispal de Braga no século XVII.

Com a utilização deste sistema, os investigadores, professores, alunos e até mesmo curiosos sobre o tema passam a ter acesso a informação detalhada sobre o *Tombo da Mitra*, uma vez que, para além de poderem consultar os fólios em formato digital, conseguem também aceder, de forma rápida e intuitiva, a um conjunto de entidades anotadas e indexadas que transmitem uma interpretação rigorosa e detalhada dos textos.

Com o objetivo de desenvolver o sistema de anotação automática de texto, foram estudadas várias técnicas de PLN, nomeadamente mecanismos de NER, que permitissem identificar e reconhecer entidades num documento textual. Após a realização deste estudo inicial, foram definidos os principais objetivos da aplicação, os módulos pelos quais iria ser composta, bem como as suas principais funcionalidades. Assim que a modelação do projeto foi dada como terminada, foi possível iniciar a fase seguinte, de desenvolvimento, que envolveu a implementação dos diferentes módulos do sistema.

A implementação do sistema de anotação iniciou-se a partir da criação de mecanismos de pré-processamento dos documentos do *Livro das Propriedades*, ou seja, foi necessário desenvolver vários mecanismos de processamento para fazer a atualização de grafia, a divisão das palavras de um texto e a classificação dessas palavras. Na criação destes dois últimos mecanismos, recorreu-se à ferramenta *LinguaKit* (Gamallo and Garcia, 2017), que possui as funcionalidades de *tokenizador* e de classificador morfológico, o que facilitou muito o seu desenvolvimento.

Em seguida, face à necessidade de anotar locais presentes nos textos, foi produzido o dicionário de localidades, a partir do processamento dos dados presentes no *dataset* (Portal de Dados Abertos da Administração Pública, 2020), que está disponível na plataforma *dados.gov.pt*. Esta etapa foi fundamental no desenvolvimento do sistema de anotação, no sentido em que tornou exequível a anotação da maior parte das localidades do território nacional. Por outro lado, o nome de algumas localidades presentes nos textos sofreu alterações ao longo dos séculos, o que fez com que fosse necessário associá-los aos respetivos nomes de localidades atuais. Aqui, o mecanismo de atualização de grafia apresentou-se como uma ferramenta essencial na identificação e na anotação de locais. Os dicionários de palavras das restantes *tags* (nomes, terrenos, profissões, etc.) foram fornecidos pela professora Anabela Barros e fazem parte do estudo lexical sobre o *Livro das Propriedades* (Barros, 2019).

Em seguida, procedeu-se ao desenvolvimento dos módulos de anotação automática e manual. O módulo de anotação automática foi responsável por identificar e anotar entidades presentes nos dicionários de palavras das respetivas *tags*, enquanto o módulo de anotação manual permitiu anotar entidades que não constavam nos dicionários de *tags*, mas que estavam presentes nos fólios, o que possibilitou a identificação de potenciais elementos que até então ainda não tinham sido descobertos.

Assim que estes módulos foram terminados e validados, procedeu-se à incorporação dos mesmos no servidor de *backend* do sistema *Tommi*. Posteriormente, foi desenvolvida uma interface capaz de disponibilizar aos utilizadores do sistema uma forma fácil, rápida e intuitiva de lidar com o sistema de anotação, através de ferramentas capazes de processar e anotar os textos do *Livro das Propriedades* que tivessem sido inseridos na plataforma. Além disso, foi desenvolvido um editor de etiquetas que oferece ao utilizador a possibilidade de alterar a anotação de uma palavra ou anotar manualmente alguma entidade que, por algum motivo, não tenha sido identificada pelo sistema. O resultado do processo de desenvolvimento materializou-se num sistema de anotação bastante versátil, provido de meios essenciais para a realização da anotação automática dos textos, mas também para o utilizador realizar uma anotação manual, caso tenha necessidade de o fazer.

Ao longo do desenvolvimento do sistema surgiram algumas dificuldades, em particular nas etapas de modelação e de implementação do projeto. Em primeiro lugar, durante a modelação do sistema, constatou-se que, apesar de ser algo comum num sistema de reconhecimento de entidades, neste caso foi bastante difícil integrar um modelo de *machine learning* supervisionado no sistema, devido à falta de dados que pudessem servir para o treino do modelo de *machine learning*. Os textos presentes no *Livro das Propriedades* possuem uma escrita característica do século XVII. Portanto, a utilização de textos atuais com o objetivo de servirem como dados de treino do modelo de aprendizagem tornou-se inviável. Por outro lado, verificou-se, também, a impossibilidade de utilizar os fólios disponíveis até

ao momento do desenvolvimento do sistema, visto que o número de documentos editados em formato digital era bastante reduzido, tendo estes sido anotados manualmente para a sua utilização como dados de treino — tarefa que seria muito morosa e requeria a intervenção de um estudioso da área.

Em segundo lugar, detetámos um outro obstáculo durante o desenvolvimento da interface do editor, que se revelou como o mais desafiador ao longo da execução de todo o projeto. A ideia inicial consistia em apresentar o texto no editor do sistema de anotação, no qual cada uma palavras corresponderia ao elemento *button*[1], isto é, por cada palavra presente no texto seria construído um elemento *button*, para que, quando se carregasse no elemento, fossem apresentadas as várias opções disponíveis (adicionar/remover/editar *tag*). Após alguns testes, verificou-se que esta solução era inviável, uma vez que apresentava péssimos resultados ao nível da performance do sistema. A segunda abordagem, que acabou por se tornar na solução final, consistiu em transformar cada palavra do texto no elemento *web* — âncora[2]. Desta forma, conseguimos apresentar as mesmas funcionalidades que no caso anterior, mas com um desempenho melhorado em termos de rapidez e eficiência.

Em contrapartida às dificuldades apresentadas, destacam-se, contudo, alguns pontos positivos encontrados ao longo do projeto. Desde logo, é necessário destacar a ferramenta *LinguaKit*, pois possui vários mecanismos de PLN compatíveis com a língua portuguesa, o que permitiu acelerar o processo de desenvolvimento do sistema de anotação de textos. Além disso, destaca-se ainda a qualidade do *dataset* extraído da plataforma *dados.gov.pt*, pois possui dados bastante completos e de fácil processamento, os quais foram fundamentais para a construção do dicionário de localidades.

Em suma, durante os trabalhos desta dissertação foram superadas várias dificuldades — encontradas quer ao nível do desenvolvimento quer da modelação do sistema — e que afetaram diretamente as decisões tomadas na concretização do projeto. Progressivamente, as restrições foram ultrapassadas e o processo de desenvolvimento foi avançando, produzindo um sistema de anotação de textos composto por um conjunto de funcionalidades interessante, bastante úteis no estudo do *Livro das Propriedades*, através da identificação de conceitos históricos.

6.2 Trabalho Futuro

O sistema de anotação de textos desenvolvido ao longo da presente dissertação, e integrado com sucesso na plataforma *Tommi*, apresenta resultados e funcionalidades bastante satisfatórios e completos.

1 https://www.w3schools.com/tags/tag_button.asp
2 https://www.w3schools.com/tags/tag_a.asp

No entanto, reconhece-se que alguns dos seus mecanismos podem ser aprimorados no futuro, por forma a melhorar a precisão do sistema de anotação. Por exemplo, é possível melhorar o módulo de *Tagging*, a partir da conceção e implementação de um mecanismo de *machine learning* supervisionado. Esta solução permitirá aumentar a precisão de anotação de entidades. Este tipo de ferramenta, quando devidamente implementada e com um vasto conjunto de dados de treino, poderá reduzir a percentagem de erros do processo de anotação de entidades e, simultaneamente, aumentar a capacidade de identificação de novos conceitos que, por alguma razão, ainda não tinham sido descobertos. Como tal, a integração de um mecanismo com esta capacidade pode melhorar substancialmente a qualidade do presente sistema de anotação de textos.

Adicionalmente, o módulo de anotação manual que foi desenvolvido pode ser aprimorado com novos identificadores de classe para cada uma das *tags*, de forma a aumentar a precisão da anotação, a partir da capacidade de identificar novos conceitos que não constem no dicionário de palavras das etiquetas. Outro exemplo semelhante é o módulo de atualização de grafia, que pode ser aperfeiçoado através do aumento do número de regras de atualização, o que permitirá realizar uma conversão mais profunda dos textos, cada vez mais próximos da norma ortográfica atual. Estes exemplos podem ser implementados no futuro, com o tempo e recursos adequados, oferecendo ao sistema *Tommi* e, consequentemente, ao sistema de anotação, mecanismos mais eficazes e precisos nas tarefas de identificação e anotação do conteúdo de textos.

BIBLIOGRAFIA

Vue.js, 2021. URL https://vuejs.org/.

Farid Ahmadi and Hamed Moradi. A hybrid method for Persian Named Entity Recognition. *2015 7th Conference on Information and Knowledge Technology, IKT 2015*, (May 2015), 2015. doi: 10.1109/IKT.2015.7288806.

Anabela Barros, Orlando Belo, João Gomes, Tiago Fraga, Ricardo Martins, and José Pedro Carvalho. A COMPUTATIONAL INSTRUMENT FOR STUDENTS ACCESSING AND EXPLORING THE BOOK OF PROPERTIES OF THE BRAGA ARCHBISHOP'S TABLE (17TH CENTURY). pages 1–7. Universidade do Minho, 2020.

Anabela Leal Barros. Apontamentos lexicais sobre o Livro das Propriedades ou Tombo da Mitra Arcebispal de Braga: designações de terras e outros aspectos das propriedades. In Imprensa da Universidade de Coimbra, editor, *Estudos de linguistica histórica: mudança e estandardização, Coimbra*, pages 393–428. Universidade de Coimbra, Coimbra, 2019.

Anabela Leal Barros. A edição do Livro das Propriedades ou Tombo da Mitra Arquiepiscopal de Braga. In *Os sete castelos. Congresso de Homenagem a D.Rodrigo de Moura Teles*, Braga, 2021.

Darina Benikova, Seid Muhie Yimam, Prabhakaran Santhanam, and Chris Biemann. GermaNER : Free Open German Named Entity Recognition Tool. 1(1):31–38, 2010.

Stephan Bloehdorn, Philipp Cimiano, and Andreas Hotho. Learning Ontologies to Improve Text Clustering and Classification. In Springer Link, editor, *From Data and Information Analysis to Knowledge Engineering*, page 8. Springer, Berlin, Heidelberg, 2006. doi: https://doi.org/10.1007/3-540-31314-1_40. URL https://link.springer.com/chapter/10.1007/3-540-31314-1_40.

Lijuan Cai and Thomas Hofmann. Text Categorization by Boosting Automatically Extracted Concepts. In ACM, editor, *SIGIR '03 Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, page 8, Toronto, Canada, 2003. doi: 10.1145/860435.860470. URL https://dl.acm.org/citation.cfm?id=860470.

Ke Chen, Lei Feng, Qingkuang Chen, Gang Chen, and Lidan Shou. EXACT: Attributed Entity Extraction By Annotating Texts. In ACM, editor, *SIGIR'19 Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*,

page 4, Paris, France, 2019. doi: 10.1145/3331184.3331391. URL https://dl.acm.org/citation.cfm?id=3331391.

Benjamin Chu, Fadzly Zahari, and Dickson Lukose. Benchmarking T-ANNE: Text Annotation System. In ACM, editor, *i-KNOW '12 Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies*, page 5, Graz, Austria, 2012. doi: 10.1145/2362456.2362464. URL https://dl.acm.org/citation.cfm?id=2362464.

Marco Cornolti, Paolo Ferragina, and Massimiliano Ciaramita. A Framework for Benchmarking Entity-Annotation Systems. In *WWW '13 Proceedings of the 22nd international conference on World Wide Web*, page 11, Pisa, Italy, 2013. University of Pisa, Italy, ACM. doi: 10.1145/2488388.2488411. URL https://dl.acm.org/citation.cfm?id=2488411.

Mariana Dias, João Boné, João C. Ferreira, Ricardo Ribeiro, and Rui Maia. Named entity recognition for sensitive data discovery in portuguese. *Applied Sciences (Switzerland)*, 10(7), 2020. ISSN 20763417. doi: 10.3390/app10072303.

Carol Porter-o Donnell and Carol Porter-o Donnell. Using the Categories to Teach Annotating. 93(5):82–89, 2004.

Erin Lynch. Annotating Text Strategies That Will Enhance Close Reading, 2021. URL https://www.sadlier.com/school.

Liliana da Silva Ferreira. *Medical Information Extraction in European Portuguese*. Phd, Universidade de Aveiro, 2011.

Mark Finlayson and Tomaay Erjavec. *Overview of Annotation Creation: Processes and Tools*, pages 167–191. 06 2017. doi: 10.1007/978-94-024-0881-2_5.

Pablo Gamallo and Marcos Garcia. LinguaKit: uma ferramenta multilingue para a análise linguística e a extração de informação. *Linguamática, 9(1)*, pages 19–28, jul 2017. doi: https://doi.org/10.21814/lm.9.1.243. URL https://linguamatica.com/index.php/linguamatica/article/view/243.

George Karypis. CLUTO, 2003.

Silvia Maria Wanderley Moraes and Vera Lucia Strube de Lima. Abordagem nao supervisionada para Extracao de Conceitos a partir de Textos. In ACM, editor, *WebMedia '08 Companion Proceedings of the XIV Brazilian Symposium on Multimedia and the Web*, page 5, Vila Velha, Espírito Santo, Brazil, 2008. doi: 10.1145/1809980.1810066. URL https://dl.acm.org/citation.cfm?id=1810066.

Lluis Padró and Evgeny Stanilovsky. FreeLing 3.0: Towards Wider Multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012) ELRA.*, page 7, Istanbul, Turkey, 2012.

Pallets. Welcome to flask, 2021. URL https://flask.palletsprojects.com/en/2.0.x/.

Portal de Dados Abertos da Administração Pública. Freguesias de Portugal - dados.gov.pt - Portal de dados abertos da Administração Pública, 2020. URL https://dados.gov.pt/pt/datasets/freguesias-de-portugal/.

Python. Python documentation, 2021. URL https://docs.python.org/3/.

Refinitiv. Elektron Data Platform, 2019. URL https://www.refinitiv.com/en/products/elektron-enterprise-data-management.

Refinitiv. Intelligent Tagging. Technical report, Refinitiv, 2020. URL https://www.refinitiv.com/content/dam/marketing/en_us/documents/fact-sheets/intelligent-tagging-fact-sheet.pdf.

Refinitiv. Intelligent Tagging helps NBC search, verify and deliver Breaking News faster than ever. Technical report, Refinitiv, 2021.

TagTop. Welcome to tagtog, 2019. URL https://docs.tagtog.net/.