CENTRO UNIVERSITÁRIO INSTITUTO DE EDUCAÇÃO SUPERIOR DE BRASÍLIA

Bacharelado em
Ciência de Dados e Inteligência Artificial

Trabalho de Conclusão de Curso - TCC

Flávia Guimarães Gaia Paula

Aplicação de técnicas de reconhecimento de entidades nomeadas em textos jurídicos: um estudo de caso

Brasília
2023

Aplicação de técnicas de reconhecimento de entidades nomeadas em textos jurídicos: um estudo de caso

Trabalho de Conclusão de Curso (TCC) apresentado como pré-requisito para obtenção de Bacharelado em Ciência de Dados e Inteligência Artificial pelo Instituto de Educação Superior de Brasília - IESB.

Orientador: Professor Sérgio da Costa Côrtes

Aplicação de técnicas de Reconhecimento de Entidades Nomeadas em textos jurídicos: um estudo de caso

Flávia Guimarães Gaia Paula

Trabalho de Conclusão de Curso (TCC) apresentado como pré-requisito para obtenção de Bacharelado em Ciência de Dados e Inteligência Artificial pelo Instituto de Educação Superior de Brasília - IESB

Brasília – Distrito Federal, em _____ de _____ de _____

Banca Examinadora constituída pelos Professores:

Prof. Dr. Sérgio da Costa Côrtes - Orientador
Professor no IESB

Prof. MSc. Alexandre Vaz Roriz - Membro
Professor no IESB

Profª. Drª. Simone de Araújo Góes Assis - Membro
Professora no IESB

Prof. Dr. Moises Fabiano Pereira da Silva Júnior - Membro
Professor no IESB

Agradecimentos

Agradeço a Deus por ter me concedido força para superar todos os obstáculos e dificuldades encontrados ao longo do desenvolvimento deste trabalho.

Gostaria de agradecer meus pais, Roberto e Maria de Fátima, pelo amor, carinho e apoio incondicional ao longo dos anos de faculdade. Agradeço por acreditarem em mim.

Ao meu irmão, Victor, agradeço por seu apoio e incentivo constantes.

Expresso minha gratidão à Professora Dra. Letícia Toledo Maia Zoby pelo apoio e orientações fornecidas durante esta pesquisa.

Ao Professor Alexandre Roriz pelas suas valiosas contribuições nesse trabalho.

Ao meu orientador Professor Sérgio da Costa Côrtes, por sua orientação, ensinamentos e disponibilidade.

Aos meus queridos amigos da faculdade, sou grata pelas palavras de motivação, força, incentivo e otimismo ao longo desta jornada acadêmica.

E agradeço a todas as pessoas que, de alguma forma, fizeram parte desta importante etapa em minha vida.

"Sem dados você é apenas mais uma pessoa com uma opinião."
(William Edwards Deming)

Resumo

A aplicação da Inteligência Artificial ao Direito tem ganhado destaque na comunidade, incluindo o Judiciário e os advogados, devido aos benefícios potenciais, como a aceleração do processo e a automação de tarefas repetitivas. Um aspecto crucial para alcançar tais objetivos é o Processamento de Linguagem Natural (PNL) , especialmente considerando que os procedimentos legais são fundamentados em documentos textuais. O reconhecimento de entidade nomeada (REN) é uma tarefa de PNL que consiste em identificar e classificar entidades nomeadas em texto não estruturado. Para aproveitar ao máximo essa abordagem, recursos de PNL em português, como modelos e conjuntos de dados, são essenciais para beneficiar os países de língua portuguesa. Este trabalho apresenta dois modelos spaCy ajustados e treinados exclusivamente em português do Brasil para a tarefa de REN no domínio jurídico. Esses modelos alcançaram o estado da arte no conjunto de dados LeNER-Br, que é um corpus REN em português voltado para o domínio jurídico. Além disso, foi desenvolvido um protótipo de aplicativo para permitir que os usuários utilizem esses modelos. Os resultados demonstraram que os modelos foram capazes de extrair informações de forma precisa e confiável. Tanto os modelos quanto o protótipo do aplicativo estão disponíveis publicamente no Hugging Face[1].

Palavras-chaves: Inteligência Artificiail, Processamento de Linguagem Natural, Direito, Reconhecimento de Entidade Nomeada. spaCy.

[1] https://huggingface.co/spaces/flaviaggp/Streamlit_Lener

ABSTRACT

The application of Artificial Intelligence to Law has gained prominence within the community, including the Judiciary and lawyers, due to potential benefits such as speeding up processes and automating repetitive tasks. A crucial aspect in achieving these objectives is Natural Language Processing (NLP), especially considering that legal procedures are grounded in textual documents. Named Entity Recognition (NER) is an NLP task that involves identifying and classifying named entities in unstructured text. To make the most of this approach, Portuguese NLP resources such as models and datasets are essential to benefit Portuguese-speaking countries. This work presents two spaCy models fine-tuned and exclusively trained in Brazilian Portuguese for the task of NER in the legal domain. These models achieved state-of-the-art performance on the LeNER-Br dataset, which is a Portuguese NER corpus specifically focused on the legal domain. Additionally, a prototype application was developed to allow users to utilize these models. The results demonstrated that the models were able to extract information accurately and reliably. Both the models and the prototype application are publicly available on Hugging Face[1].

Keywords: Artificial Intelligence, Natural Language Processing, Law, Named Entity Recognition, spaCy.

[1] https://huggingface.co/spaces/flaviaggp/Streamlit$_{L}ener$

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AM	Aprendizado de Máquina (*Machine Learning*)
API	Interface de Programação de Aplicativos (*Application Programming Interface*)
APP	Aplicativo (*Application*)
BERT	Bidirectional Encoder Representation from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
BOs	Boletins de Ocorrências
CD	Ciência de Dados (*Data Science*)
CNJ	Conselho Nacional de Justiça
CNN	Convolutional Neural network
CoNLL	Conference of Natural Language Learning
CRF	Conditional Random Field
DSRPAI	Dartmouth Summer Research Project on Artificial Intelligence
FN	False Negatives
FP	False Positives
Human NERD	Human Named Entity Recognition with Deep learning
IA	Inteligência Artificial (*Artificial Intelligence*)
LOC	Localização
LTSM	Long Short Term Memory
ORG	Organização
PER	Pessoa
PLN	Processamento de Linguagem Natural (*Natural Language Processing*)
REN	Reconhecimento de Entidade Nomeada (*Named Entity Recognition*)
RNA	Redes Neurais Artificiais

TN	True Negatives
TP	True Positives

Sumário

1 INTRODUÇÃO

1.1 Contextualização

A Inteligência Artificial tem sido uma aliada no meio jurídico, por meio de técnicas estabelecidas, principalmente na área de Processamento de Linguagem Natural. Isso se deve ao fato de que a maioria dos dados jurídicos, produzidos tanto dentro quanto fora dos tribunais, são apresentados em textos não estruturados, em sistemas ou em documentos (DALE, 2019). Além do ganho de tempo na criação de peças jurídicas proporcionado pelos meios digitais, a popularização das aplicações de inteligência artificial está abrindo caminho para o acesso a informações que auxiliam na tomada de decisões dos agentes jurídicos (MOTA et al., 2021).

O reconhecimento de entidades nomeadas é uma das tarefas fundamentais do PLN e tem sido amplamente aplicado em diversas áreas, como em sistemas de recomendação, análise de sentimentos e chatbots, entre outros. Em documentos jurídicos, o REN é uma técnica crucial para extrair informações precisas e relevantes, como nomes de pessoas, organizações, locais, datas e valores monetários (GONCALVES, 2017).

No entanto, a análise de documentos jurídicos é uma tarefa desafiadora devido à complexidade da linguagem utilizada, bem como à diversidade de documentos jurídicos existentes, como contratos, petições, acórdãos, entre outros. Além disso, o português, sendo uma das línguas mais faladas no mundo, possui uma rica diversidade linguística, o que torna o processo de REN ainda mais complexo (SILVA, 1988).

Nesse contexto, os modelos pré-treinados do spaCy têm sido amplamente utilizados para o REN. Entretanto, há uma escassez de estudos que comparem o desempenho de diferentes modelos de spaCy para o REN em documentos jurídicos em português.

Dessa forma, este trabalho visa realizar uma análise comparativa de dois modelos spaCy para o REN em documentos jurídicos em português. O resultado desse estudo pode ser útil para profissionais do setor jurídico que precisam analisar documentos de forma mais precisa e eficiente. Além disso, o estudo contribui para a literatura de PLN em português, fornecendo informações úteis sobre o desempenho de modelos spaCy para o REN em textos legais.

1.2 Motivação

A motivação para este trabalho de conclusão de curso baseia-se nos dados alarmantes apresentados no mais recente Relatório Justiça em Números, elaborado pelo Departamento

de Pesquisas Judiciárias do Conselho Nacional de Justiça (CNJ), que é reconhecido como a principal fonte de medição da atividade judicial. De acordo com esse relatório, o Brasil enfrenta um número extenso de processos pendentes, que atingiu a marca de 77,3 milhões em 2021, abrangendo todos os ramos de justiça e todos os tribunais do país(JUSTIçA, 2022).

Além disso, constatou-se um aumento de 1,85% no número de processos pendentes em relação ao ano anterior, quando já havia 75,9 milhões de casos pendentes (JUSTIçA, 2022), como ilustra o gráfico da Figura 1. Esse crescimento preocupante pode ser atribuído a diversos fatores, incluindo o excesso de demandas e a falta de estrutura enfrentada pelo Judiciário brasileiro (REINA, 2022).

Figura 1 – Série histórica dos casos pendentes

Fonte: (JUSTIçA, 2022)

Diante desse cenário, torna-se crucial buscar soluções que possam agilizar e otimizar o processo judicial, visando reduzir o número de processos pendentes. Nesse contexto, o presente trabalho propõe a aplicação de técnicas de REN em documentos jurídicos como uma abordagem promissora. Essas técnicas têm o potencial de facilitar a análise e extração de informações relevantes, contribuindo para uma análise mais precisa e eficiente dos documentos jurídicos.

Ao explorar essas técnicas de REN, espera-se que seja possível automatizar parte do processo de análise documental, agilizando o trabalho dos profissionais do setor jurídico e proporcionando uma tomada de decisão mais ágil. Com isso, espera-se contribuir para a redução dos processos pendentes, aliviando a carga de trabalho dos tribunais e promovendo uma maior eficiência na administração da justiça.

1.3 Objetivos

O objetivo geral deste trabalho é implementar dois modelos baseados em técnicas de Inteligência Artificial para auxiliar no reconhecimento de entidade nomeada aplicado

ao domínio jurídico. Para alcançar este objetivo, os objetivos específicos propostos neste trabalho se configuram em:

• Comparar o desempenho de dois modelos pré-treinados do spaCy para o REN em textos jurídicos em português.

• Analisar a influência do tamanho do modelo (small vs large) no desempenho do REN em documentos jurídicos em português.

• Construir um Web APP para que os usuário possam testar os modelos.

1.4 Conteúdo e organização

Este trabalho é estruturado em cinco capítulos principais. No Capítulo 2, é apresentada a fundamentação teórica, que forneceu a base necessária para o desenvolvimento deste estudo. No Capítulo 3, são apresentados cinco estudos relacionados que contribuíram para o contexto atual do trabalho. O Capítulo 4 descreve detalhadamente a implementação do projeto, abrangendo os materiais, métodos e os dados utilizados. Os resultados obtidos são apresentados no Capítulo 5. Por fim, no Capítulo 6, são apresentadas as conclusões deste estudo, destacando os principais resultados alcançados e oferecendo sugestões para trabalhos futuros.

2 Fundamentação teórica

Neste capítulo, serão expostos conceitos fundamentais sobre as técnicas de IA, bem como as métricas de avaliação a serem empregadas neste trabalho.

2.1 Inteligência Artificial

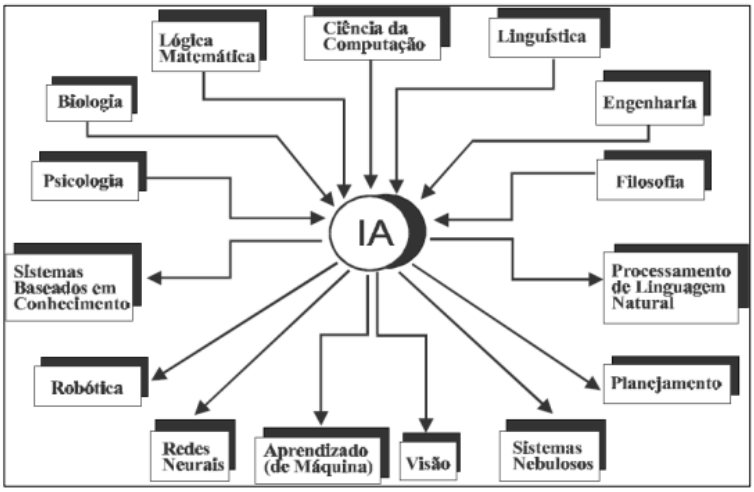
Segundo (MCCARTHY, 2007), a questão fundamental que originou o campo da Inteligência Artificial foi a possibilidade de as máquinas serem capazes de pensar, assim como os seres humanos. Alan Turing foi o responsável por levantar essa questão e, na década de 1950, publicou o artigo "Computing Machinery and Intelligence"na revista Mind, que é considerado pioneiro no campo da IA. Nesse artigo, ele explorou a possibilidade de instruir as máquinas a pensar, entender, aprender e aplicar sua própria "inteligência"na resolução de problemas. Além disso, o trabalho introduziu o conceito do Teste de Turing, que é utilizado para medir a capacidade de uma máquina exibir comportamento inteligente equivalente ao de um ser humano.

Antes da década de 1950, os computadores não tinham a capacidade de armazenar comandos, apenas de executá-los, o que era um pré-requisito essencial para o desenvolvimento da IA. Foi somente em 1956 que o primeiro projeto de IA foi apresentado: The Logic Theorist, criado pelos pesquisadores americanos Allen Newell e Herbert A. Simon. Esse programa foi projetado para imitar as habilidades de resolução de problemas humanas e foi apresentado no Dartmouth Summer Research Project on Artificial Intelligence, um workshop de verão organizado pela Faculdade Dartmouth (EUA), que é considerado o evento fundador da IA como campo de estudo (ANYOHA, 2017).

A IA é uma área de estudo que se dedica a criar máquinas inteligentes, em especial programas de computador inteligentes. O objetivo da IA é entender a inteligência humana e reproduzi-la em máquinas. Em outras palavras, a IA é a capacidade de um computador digital ou de um robô controlado por computador para executar tarefas que geralmente requerem habilidades associadas a seres inteligentes (MCCARTHY, 2007).

Devido à sua amplitude, a IA está relacionada com diversas áreas científicas, como psicologia, biologia, lógica matemática, linguística, engenharia e filosofia, entre outras, conforme ilustrado na Figura 2.

Figura 2 – Áreas Relacionadas à Inteligência Artificial



Fonte: (MONARD; BARANAUKAS, 2000).

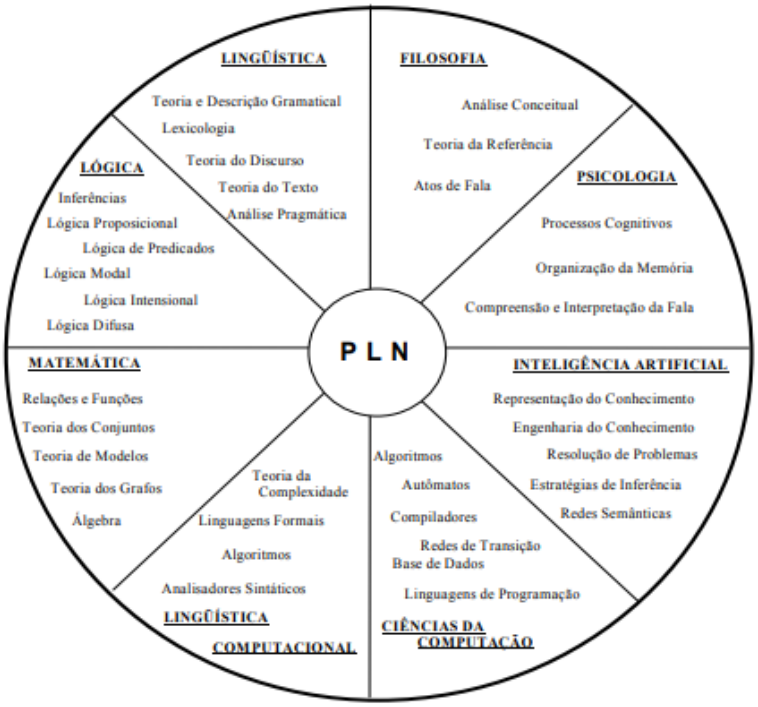
A área de IA tem se desenvolvido em diversas linhas de pesquisa, tais como Sistemas Baseados em Conhecimento, Robótica, Redes Neurais, Aprendizado de Máquina, Visão Computacional, Lógica Nebulosa, Planejamento, Processamento e Interpretação de Linguagem Natural, Reconhecimento de Padrões, entre outras, com o objetivo de capacitar o computador a realizar funções que antes só eram possíveis através da inteligência humana (MONARD; BARANAUKAS, 2000).

2.2 Processamento de Linguagem Natural

O Processamento de Linguagem Natural começou na década de 1940, após a Segunda Guerra Mundial, quando as pessoas procuravam criar uma máquina capaz de traduzir automaticamente um idioma para outro. No entanto, os desafios eram mais complexos do que imaginavam. Em 1958, Noam Chomsky identificou questões significativas no desenvolvimento da PNL, especialmente a capacidade de reconhecer sentenças gramaticalmente corretas, mas sem sentido. A partir de então, os pesquisadores se dividiram em duas divisões: simbólica e estocástica. Posteriormente, novas áreas surgiram, como a compreensão da linguagem natural e a modelagem do discurso. Nos anos 80 e 90, a PNL se tornou mais unida no foco no empirismo e em modelos probabilísticos (ROBERTS, 2009).

O PLN é uma área da IA que tem como objetivo compreender como as máquinas podem compreender e produzir a linguagem natural (JURAFSKY; MARTIN, 2008). O estudo do PLN é um campo de pesquisa amplo e produtivo que se relaciona com várias disciplinas. A Figura 3 apresenta uma sistematização dos principais recursos teórico-metodológicos disponíveis para o estudo do PLN (SILVA, 2006).

Figura 3 – Recursos teórico-metodológicos para o estudo do PLN



Fonte: (SILVA, 2006).

(JURAFSKY; MARTIN, 2008) retratam que o PLN está amplamente ligado com o estudo da lingua e seus conceitos mais complexos, dentre as áreas da linguística que apresentam forte conexão com o PLN, destacam-se aquelas que envolvem um comportamento linguístico complexo, demandando diversos tipos de conhecimento da língua, como:

- Fonética e Fonologia, que envolvem o conhecimento dos sons linguísticos;
- Morfologia, que compreende o conhecimento dos componentes significativos das palavras;
- Sintaxe, que engloba o aprendizado das relações estruturais entre as palavras;
- Semântica, que inclui o familiaridade do significado;
- Pragmática, que envolve o conhecimento da relação entre o significado e os objetivos e intenções do falante;
- Discurso, que está ligado ao entendimento sobre unidades linguísticas maiores do que um único enunciado.

Além disso, segundo (SHANMUGAMANI; ARUMUGAM, 2018), o PNL apresenta uma ampla variedade de aplicações, tais como:

Análise de sentimentos , que consiste em identificar a opinião positiva, negativa ou neutra de um texto, quantificando-a através de valores discretos ou em escala contínua;

Reconhecimento de entidades nomeadas , que rotula palavras ou tokens em categorias específicas, convertendo dados não estruturados em dados estruturados para análises futuras;

Vinculação de entidades , que extrai relacionamentos úteis entre entidades; tradução automática, que traduz um trecho de texto de um idioma para outro;

Tradução de texto , na tarefa de traduzir um determinado trecho de texto de um idioma para outro idioma de destino

Inferência de Linguagem Natural é a tarefa de determinar se a dada "hipótese" segue logicamente da "premissa". Em termos leigos, você precisa entender se a hipótese é verdadeira, enquanto a premissa é o seu único conhecimento sobre o assunto.

Extração de relação , que prevê uma relação quando um texto e um tipo de relação são fornecidos;

Geração de consulta SQL ou Análise semântica , que ajuda a converter linguagem natural em consultas SQL para consultar um banco de dados;

Compreensão de máquina (MC) , que responde perguntas de um parágrafo;

Envolvimento textual , que prevê se os fatos em diferentes textos são iguais; resolução de correferência, que resolve pronomes em um texto quando várias pessoas interagem;

Resolução de correferência , a resolução pronominal resolve os pronomes em um texto quando há várias pessoas interagindo;

Busca de informações , é parte integrante do acesso à Internet e é uma aplicação da PNL, como os serviços de pesquisa sãofornecidos pela API do Bing;

Chatbots , que geram respostas a partir de um contexto fornecido com uma pergunta;

Conversão de texto para voz e de voz para texto , em que um texto precisa ser convertido em voz. Pode ser útil para um bot pessoal responder a um usuário.

Identificação do locutor , que encontra o nome da pessoa que está falando;

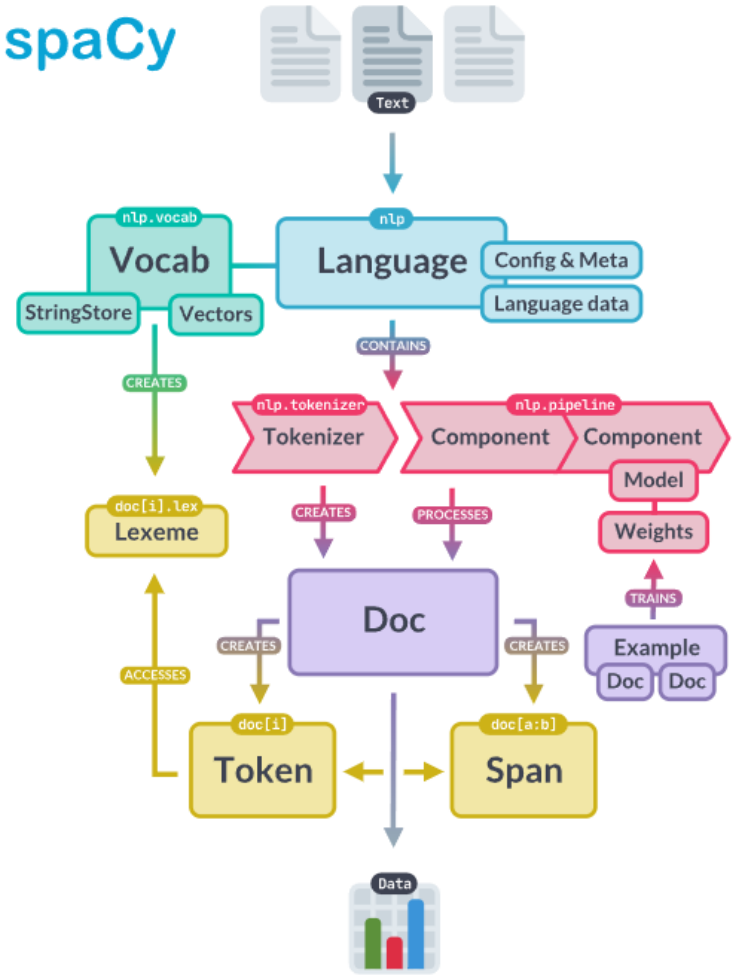
Sistemas de diálogo falado , que combinam diferentes aplicativos para formar uma experiência em sistemas de diálogo.

Outras aplicações incluem a detecção de spam e a classificação de notícias.

2.2.1 SpaCy

SpaCy é uma biblioteca em Python para PLN em escala industrial. Ela oferece diversos recursos para análise textual, baseados em modelos estatísticos e regras linguísticas. Com ela, é possível realizar tarefas como tokenização, lematização, marcação de parte do discurso, REN e análise sintática (SPACY, 2023). A biblioteca SpaCy simplifica a implementação de tarefas de PLN em diferentes aplicações, graças à sua arquitetura eficiente e aos seus modelos pré-treinados de alta qualidade. Ela também possui uma ampla gama de funcionalidades e é uma opção popular entre pesquisadores e desenvolvedores (SPACY, 2023).

Figura 4 – Arquitetura da biblioteca spaCy



Fonte: Elaborada por (SPACY, 2023).

A arquitetura dessa biblioteca, apresentada na Figura 4, baseia-se em três estruturas de dados centrais: a classe Language, o objeto Vocab e o objeto Doc. A classe Language é utilizada para processar um texto e transformá-lo em um objeto Doc. Geralmente, essa classe é armazenada em uma variável chamada "nlp". O objeto Doc é responsável por armazenar a sequência de tokens do texto e todas as suas anotações. Ao centralizar strings, vetores de palavras e atributos lexicais no objeto Vocab, evita-se armazenar cópias

múltiplas desses dados, o que economiza memória e garante que haja apenas uma única fonte confiável dessas informações (SPACY, 2023).

As anotações de texto também são projetadas para permitir uma única fonte confiável: o objeto Doc é o proprietário dos dados, enquanto os objetos Span e Token são visualizações que apontam para ele. O objeto Doc é construído pelo Tokenizer e posteriormente modificado pelos componentes do pipeline de processamento. O objeto Language coordena esses componentes. Ele recebe o texto original, o envia pelo pipeline de processamento e retorna um documento anotado. Além disso, o objeto Language também gerencia o treinamento e a serialização do modelo (SPACY, 2023), como mostrado na Figura 4.

Os modelos do spaCy suportam totalmente 24 idiomas e outros 48 parcialmente, sendo pré-treinados para tarefas como tokenização, marcação, análise, lematização e reconhecimento de entidade nomeada (SHIRALY, 2023). Dentre os diversos modelos pré-treinados disponibilizados, estão os modelos "pt_core_news_sm"e "pt_core_news_lg", desenvolvidos especificamente para a língua portuguesa. Esses modelos contêm vocabulário, sintaxe e informações sobre entidades para análise de textos escritos, como notícias e mídia. A diferença entre eles reside principalmente no tamanho e nos vetores de palavras. O modelo "sm"é mais compacto e rápido, ocupando apenas 12 MB, porém é ligeiramente menos preciso. Já o modelo "lg"é maior, com 541 MB, e mais lento, mas apresenta maior precisão (SPACY, 2023).

O processo de ajuste fino no spaCy consiste em treinar modelos específicos para tarefas mais especializadas. Por exemplo, é viável realizar o ajuste fino de um modelo pré-treinado do spaCy para realizar a classificação de sentimentos ou reconhecer entidades específicas em um determinado domínio textual (SPACY, 2023).

Em síntese, spaCy é uma biblioteca poderosa e flexível para o processamento de linguagem natural, que possibilita a execução eficiente e precisa de diversas tarefas. A disponibilidade de modelos pré-treinados e a capacidade de ajuste fino tornam o spaCy uma ferramenta valiosa para a análise e extração de informações em textos de diferentes idiomas.

2.2.2 Reconhecimento de Entidade Nomeada

Named entity recognition, ou reconhecimento de entidade nomeada, é uma tarefa que consiste em identificar e classificar palavras ou frases em um texto de acordo com classes predefinidas para o modelo REN (NADEAU; SEKINE, 2007). As classes tradicionais incluem Pessoa, Localização, Organização e um tipo diverso que abrange outros tipos não necessariamente presentes nesse conjunto convencional de classes (KONKOL; BRYCHCíN; KONOPíK, 2015).

O REN é baseado em duas atividades principais: a identificação de tokens em um texto não estruturado e a classificação desses tokens em tipos de entidades definidos com base nas características do domínio em questão (SPECK; NGOMO, 2014). Além do conjunto de entidades predefinidas, é possível adicionar um conjunto adicional de entidades nomeadas ao REN para adaptá-lo a um contexto específico. Por exemplo, no domínio jurídico, a legislação e a jurisprudência são entidades específicas desse domínio.

A escolha das entidades de interesse depende diretamente do tipo de texto no qual o modelo REN será aplicado, variando de acordo com a aplicação, podendo incluir, por exemplo, entidades relacionadas a medicamentos, doenças e outros termos biomédicos (NADEAU; SEKINE, 2007).

Em geral, o REN é utilizado em situações nas quais é útil ter uma visão geral e de alto nível de uma grande quantidade de textos. Um modelo de REN facilita as tarefas de compreender os assuntos abordados em documentos e agrupar conjuntos de textos com base em relevância ou similaridade (MARSHALL, 2019).

Os nomes das entidades são o conteúdo principal de um documento. O REN é um passo intermediário para a extração e gerenciamento ágil de informações. Embora seja relativamente simples construir um sistema REN com desempenho razoável, ainda é desafiador adaptá-lo a um domínio específico e obter um desempenho ótimo (MA; XIA, 2014).

2.3 Aprendizado de Máquina

A Aprendizagem de Máquina (AM) é uma subdivisão da Inteligência Artificial que utiliza um conjunto de ferramentas computacionais sofisticadas e autônomas para adquirir conhecimento. Essas ferramentas são menos dependentes da intervenção humana e foram desenvolvidas para lidar com a crescente complexidade dos problemas que requerem soluções computacionais, bem como o aumento da velocidade e do volume de dados gerados (FACELI et al., 2021).

A AM se baseia na ideia de que os sistemas podem absorver informações, identificar padrões e tomar decisões com pouco envolvimento humano. Em resumo, um algoritmo de ML utiliza um conjunto de dados e, com base nas correlações observadas, gera resultados. Em vez de criar rotinas de software manualmente, com um conjunto específico de comandos para executar uma tarefa específica, a máquina é "treinada"com uma grande quantidade de dados e algoritmos que permitem que ela aprenda como realizar a tarefa (COELHO, 2022).

Os métodos de aprendizado mais populares são (INSIGHTS, 2021):

- Aprendizado supervisionado: é uma técnica de AM em que o algoritmo é treinado

com exemplos previamente rotulados, ou seja, exemplos nos quais as saídas corretas são conhecidas. O objetivo do treinamento é fazer com que o algoritmo aprenda a mapear as entradas para as saídas corretas, de forma a ser capaz de generalizar para novos exemplos. No caso do REN, o algoritmo precisa aprender a mapear os textos para as entidades nomeadas corretas, com base nos exemplos previamente anotados.

- Aprendizado não-supervisionado: o treinamento é realizado sem dados rotulados. Isso significa que o algoritmo deve explorar os dados e encontrar alguma estrutura ou padrão. Exemplos incluem agrupamento k-means e mapas auto-organizáveis.
- Aprendizado semi-supervisionado: é utilizado para as mesmas aplicações do aprendizado supervisionado, mas manipula tanto dados rotulados quanto não rotulados no treinamento. Isso é útil quando o custo associado à rotulação é muito alto para permitir um processo de treinamento totalmente rotulado.
- Aprendizado por reforço: o algoritmo descobre, por meio de tentativa e erro, quais ações resultam em maiores recompensas. Esse tipo de aprendizado envolve um agente (aprendiz ou tomador de decisão), um ambiente (com o qual o agente interage) e ações (o que o agente pode fazer). O objetivo é que o agente escolha ações que maximizem a recompensa total em um determinado período de tempo. A política de recompensas é definida pelo programador. O agente alcançará o objetivo da melhor e mais rápida maneira possível se seguir um bom caminho e tomar boas decisões. O aprendizado por reforço é comumente usado em jogos, robótica e navegação.

O REN é considerado um problema de aprendizado supervisionado, uma vez que o algoritmo de REN requer exemplos previamente anotados para o treinamento. Ou seja, um conjunto de textos em que as entidades já foram identificadas e classificadas por um especialista humano. A partir desses exemplos, o algoritmo é capaz de aprender a identificar e classificar novas entidades em textos não vistos anteriormente. Portanto, o REN é um exemplo de aplicação prática do aprendizado supervisionado, em que a técnica de REN é utilizada para extrair informações importantes de textos de forma automatizada(BIRD; KLEIN; LOPER, 2009).

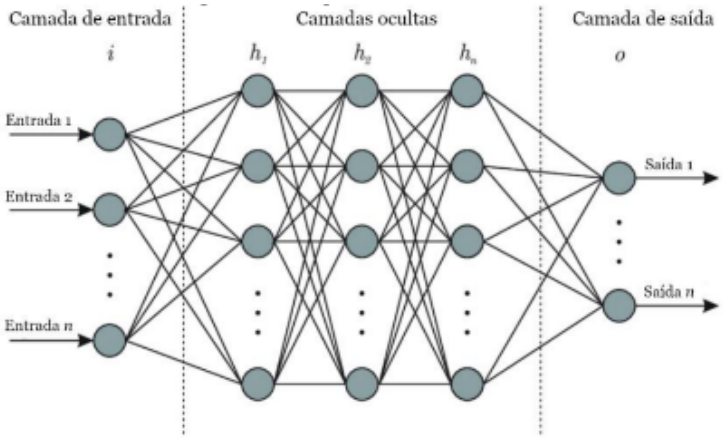
2.3.1 Redes Neurais Artificiais

Em meio as áreas de pesquisa da IA uma das mais relevantes é a simulação de capacidades cognitivas de um ser humano, em que máquinas projetadas são capazes de exibir um comportamento inteligente, bem como reações humanas. A partir desses fatos, surgiu a tentativa de reproduzir a estrutura e o funcionamento do cérebro em um ambiente técnico. Isso significa que a partir de estudos tenta-se entender o funcionamento da inteligência residente nos neurônios e mapeá-la para uma estrutura artificial, tornando assim, as redes neurais biológicas em Redes Neurais Artificiais (RNAs) (RAUBER, 2005).

Uma RNA é um sistema de processamento massivamente paralelo, composto por unidades simples com capacidade natural de armazenar conhecimento e disponibilizá-lo para uso. Em sua forma mais geral, uma rede neural é um sistema projetado para modelar a maneira como o cérebro realiza uma tarefa particular, sendo normalmente implementada utilizando-se de componentes eletrônicos ou é simulada por propagação em um computador digital. Para alcançarem bom desempenho, as redes neurais empregam uma interligação maciça de células computacionais simples, denominadas de “neurônios” ou unidades de processamento (HAYKIN, 2001).

A Figura 5 demostra como uma RNA é organizada, constituída de uma camada de entrada, uma camada de saída e zero ou mais camadas ocultas. (HAYKIN, 2009) estabelece que a camada de entrada é responsável por receber os valores iniciais. A camada de saída apresenta o resultado final produzido pelo sistema. As camadas ocultas são usadas para suceder transformações nos dados de forma que podem ser utilizados pela camada de saída para efetuar uma predição (LIMA, 2021).

Figura 5 – Arquitetura de uma RNA

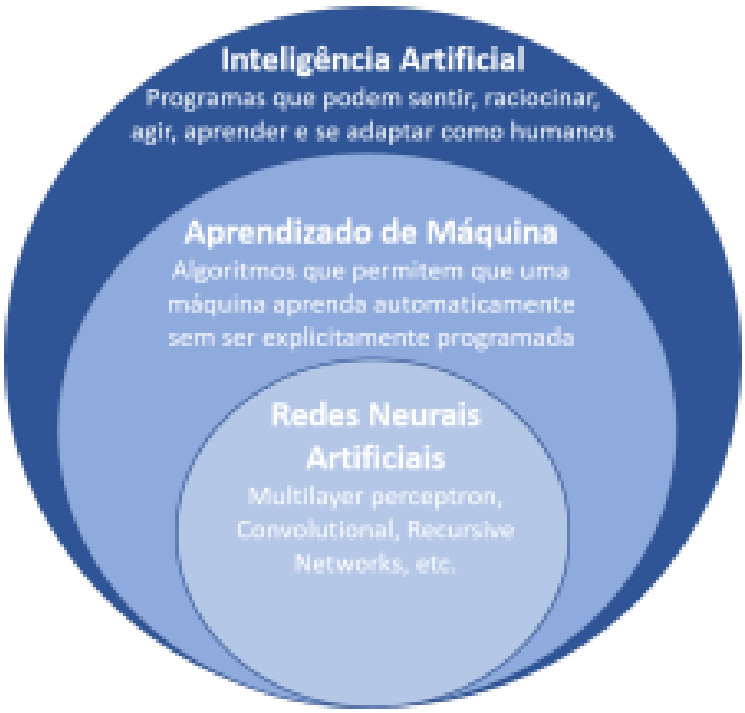


Fonte: (LIMA, 2021)

Dentre as RNAs estão as Redes Neurais Convolucionais que é o pressuposto do framework utilizado neste trabalho e por conseguinte será definida abaixo. As CNNs são especialmente úteis para o REN, pois são capazes de aprender a identificar padrões de palavras que geralmente estão associados a uma determinada entidade. Por exemplo, uma CNN poderia aprender que palavras como ”advogado”, ”juiz”e ”tribunal”geralmente estão associadas a uma entidade de tipo ”organização jurídica”.

Em síntese, as redes neurais estão ligadas ao AM e ao REN por serem capazes de aprender a identificar padrões e extrair características relevantes dos textos, permitindo que as entidades sejam identificadas de forma eficiente e automatizada. A Figura 6, que resume o relacionamento entre IA, AM e as RNAs, bem como suas definições.

Figura 6 – Diagrama: IA, AM e RNAs



Fonte: (FIGUEIREDO,)

2.3.2 Redes Neurais Convolucionais

Redes Neurais Convolucionais são uma arquitetura neural que utiliza a técnica de convolução em pelo menos uma de suas camadas, substituindo a multiplicação matricial (GUIMARAES, 2021). Essas redes foram desenvolvidas com o propósito de processar informações organizadas em uma estrutura bidimensional, como imagens e séries temporais, por meio do uso de camadas convolucionais. O uso de CNNs é amplamente adotado para o reconhecimento de padrões em imagens (WIKIPEDIA, 2020).

As CNNs diferem de outras redes neurais em sua arquitetura e operação. A principal diferença reside na utilização da camada convolucional, que é especialmente projetada para processar dados estruturados em uma grade, como imagens e séries temporais (MANDELLI, 2023). Enquanto nas redes neurais tradicionais todas as conexões entre as camadas são totalmente conectadas (cada neurônio está conectado a todos os neurônios da camada subsequente), as CNNs utilizam a operação de convolução para realizar a aprendizagem de características localmente. Essa operação de convolução envolve a aplicação de um filtro ou kernel sobre a entrada, realizando uma operação de multiplicação e soma ponderada dos valores da região correspondente (DATAPEAKER, 2021).

A maioria das arquiteturas de CNN é composta por camadas convolucionais, camadas de pooling, camadas totalmente conectadas (ou densas) e, às vezes, camadas de normalização. Essas camadas são organizadas em uma sequência específica para extrair características e tomar decisões. Essa abordagem é benéfica para o processamento de

dados com estrutura espacial, pois permite a extração de características locais relevantes, capturando informações como bordas, texturas e padrões. Além disso, as CNNs também empregam camadas de pooling para reduzir a dimensionalidade dos dados, ajudando a capturar informações importantes em diferentes escalas (GOODFELLOW; BENGIO; COURVILLE, 2016).

As CNNs possuem várias arquiteturas comumente utilizadas, cada uma com suas características específicas. Alguns exemplos populares de arquiteturas de CNNs incluem LeNet-5, AlexNet, VGGNet, GoogLeNet (Inception), ResNet e MobileNet. Cada uma dessas arquiteturas foi desenvolvida para resolver desafios específicos em termos de eficiência computacional, precisão e capacidade de aprendizado(DAS, 2017). Cada uma dessas arquiteturas foi desenvolvida para resolver desafios específicos em termos de eficiência computacional, precisão e capacidade de aprendizado.

De acordo com (GOODFELLOW; BENGIO; COURVILLE, 2016), as aplicações das CNNs são vastas e abrangem diversas áreas. Algumas das principais aplicações incluem:

Reconhecimento de imagens: As CNNs são amplamente utilizadas em tarefas de classificação de imagens, detecção de objetos, segmentação semântica e reconhecimento facial.

Processamento de vídeo: As CNNs são aplicadas no processamento de sequências de vídeo, como reconhecimento de ações, rastreamento de objetos e análise de movimento.

Processamento de áudio: As CNNs podem ser usadas em tarefas de classificação de áudio, como reconhecimento de fala, identificação de gênero e reconhecimento de música.

Visão computacional: As CNNs são utilizadas em aplicações de visão computacional, como detecção e reconhecimento de objetos em tempo real, estimação de pose e análise de cena.

Processamento de linguagem natural: As CNNs também têm sido empregadas em tarefas de processamento de linguagem natural, como classificação de texto, análise de sentimentos e tradução automática.

Essas são apenas algumas das muitas áreas em que as CNNs têm se destacado, demonstrando sua eficácia na análise de dados estruturados em grade e na extração de características relevantes para uma ampla gama de aplicações.

2.4 Métricas de avaliação

Nesta seção, serão apresentadas as métricas utilizadas para avaliar o desempenho do sistema de REN desenvolvido. Serão abordadas as métricas de precisão, recall e F1-Score, que são amplamente utilizadas na avaliação de tarefas de classificação e identificação de entidades.

As expressões matemáticas utilizadas para calcular cada uma das métricas dependem de três variáveis. No contexto do REN, elas são definidas da seguinte forma(HECK, 2022):

True Positive (TP): Representa uma entidade que é corretamente localizada e identificada. False Positive (FP): Refere-se a um termo que não é de uma determinada entidade, mas é erroneamente rotulado como tal. False Negative (FN): Indica um termo que pertence a uma determinada entidade, mas não é rotulado corretamente como tal. A seguir, serão apresentadas as definições das três métricas.

2.4.1 Precisão

A precisão é uma métrica que mede a proporção de entidades nomeadas corretamente identificadas em relação ao total de entidades nomeadas identificadas pelo sistema. Ela indica a capacidade do sistema de REN em evitar falsos positivos, ou seja, identificar erroneamente uma palavra ou expressão como uma entidade nomeada. A fórmula para o cálculo da precisão é dada ela Equação 2.1:

$$Precisão = \frac{TP}{TP + FP} \tag{2.1}$$

Valores mais altos de precisão indicam uma menor taxa de falsos positivos.

2.4.2 Recall

O recall, também conhecido como revocação, é uma métrica que mede a proporção de entidades nomeadas corretamente identificadas em relação ao total de entidades nomeadas presentes nos dados de referência. Ela indica a capacidade do sistema de REN em evitar falsos negativos, ou seja, identificar corretamente todas as entidades nomeadas presentes nos textos. A fórmula para o cálculo do recall é dada pela Equação 2.2:

$$Recall = \frac{TP}{TP + FN} \tag{2.2}$$

Valores mais altos de recall indicam uma menor taxa de falsos negativos.

2.4.3 F1-score

O F1-Score é uma métrica que combina a precisão e o recall em uma única medida, fornecendo uma medida geral de desempenho do sistema de REN. É calculado como a média harmônica entre a precisão e o recall e fornece uma visão equilibrada entre essas duas métricas. O F1-Score é especialmente útil quando há um desequilíbrio entre as classes de entidades nomeadas. A fórmula para o cálculo do F1-Score é dada pela Equação 2.3:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{2.3}$$

Idealmente, almeja-se que um modelo de REN demonstre elevados índices tanto de precisão quanto de recall, ou seja, espera-se que o algoritmo identifique a maior quantidade possível de entidades de referência, sem incluir falsos positivos. Entretanto, em certos casos, o aumento de uma dessas métricas pode ocorrer às custas da redução da outra.

3 Trabalhos Relacionados

Este capítulo expõe alguns estudos correlatos, ressaltando as analogias e disparidades em relação ao desenvolvimento efetuado nesta pesquisa.

3.0.1 LeNER-Br: A Dataset for Named Entity Recognition in Brazilian Legal Text

Existem diversos domínios que oferecem grandes volumes de textos que requerem o processo de extração de entidades nomeadas. Por exemplo, no estudo realizado por (ARAUJO, 2018), um conjunto de dados é proposto para o reconhecimento de entidades nomeadas em documentos legais brasileiros. Esse conjunto de dados consiste exclusivamente de documentos legais com marcações para pessoas, locais, entidades temporais, organizações, além de tags específicas para entidades legais e casos legais.

Além disso, (ARAUJO, 2018) também conduziu experimentos utilizando outro conjunto de dados em língua portuguesa, chamado Paramopama, que resultaram em melhorias comparadas aos relatórios anteriores, empregando a abordagem LSTM-CRF. Em seguida, o LSTM-CRF foi reajustado utilizando o conjunto de dados proposto, alcançando pontuações F1 de 97,04% e 88,82% para entidades de Legislação e Jurispredência, respectivamente. Esses resultados demonstram a viabilidade do conjunto de dados proposto para aplicações no contexto jurídico.

A principal semelhança entre o estudo realizado por (ARAUJO, 2018) e este trabalho reside na utilização do mesmo conjunto de dados para a tarefa de REN. Ambos os estudos buscam identificar entidades nomeadas em domínios similares. A diferença entre o presente trabalho e o estudo de (ARAUJO, 2018) é que este trabalho propõe a utilização da ferramenta spaCy para o processo de extração de entidades nomeadas, enquanto (ARAUJO, 2018) apresenta um modelo baseado em LSTM-CRF.

3.0.2 Reconhecimento de entidades nomeadas em documentos jurídicos em português utilizando redes neurais

No artigo de (MOTA et al., 2021), é apresentada uma abordagem para a detecção de entidades legais através da aplicação de modelos baseados em redes neurais disponíveis em bibliotecas de processamento de linguagem natural. (MOTA et al., 2021) investiga o uso das bibliotecas spaCy e FLAIR no contexto de REN em petições iniciais, avaliando-as em dois corpora, sendo um deles desenvolvido durante o próprio trabalho e o outro desenvolvido por (ARAUJO, 2018). Os resultados obtidos demonstraram um desempenho promissor

tanto com a plataforma spaCy quanto com a FLAIR, sendo que o modelo BiLSTM-CRF com FLAIR embeddings obteve um desempenho superior.

O estudo realizado por (MOTA et al., 2021) apresenta semelhanças com o presente trabalho no uso da biblioteca spaCy e do conjunto de dados mencionado em (ARAUJO, 2018) para realizar a tarefa de REN. No entanto, a diferença reside na utilização de uma versão mais recente da biblioteca spaCy neste trabalho, e também na utilização de apenas um conjunto de dados mencionado em (ARAUJO, 2018).

3.0.3 Fostering Judiciary Applications with New Fine-Tuned Models for Legal Named Entity Recognition in Portuguese

O trabalho realizado por (ZANUZ; RIGO, 2022) apresenta os primeiros modelos BERT ajustados, treinados exclusivamente em português do Brasil, para a tarefa de REN no contexto jurídico. No estudo, eles utilizaram o conjunto de dados LeNER-Br. Além disso, foi desenvolvido um protótipo de aplicativo destinado aos usuários do Judiciário para avaliar o desempenho do modelo com documentos de lei autênticos. Os resultados obtidos demonstraram que os modelos foram capazes de extrair informações com alta qualidade.

O presente trabalho compartilha pontos em comum com o estudo realizado por (ZANUZ; RIGO, 2022), tanto no uso do conjunto de dados LeNER-Br quanto na criação de um protótipo de aplicativo para que os usuários possam avaliar o desempenho dos modelos.

3.0.4 Reconhecimento de entidades nomeadas em textos de boletins de ocorrência

O trabalho desenvolvido por(ARAÚJO, 2019) introduz uma ferramenta interativa projetada para auxiliar o usuário nas tarefas de classificação REN, abrangendo desde a criação de um amplo conjunto de dados até a criação/manutenção de um modelo REN de Deep Learning chamado Human NERD. Além disso, essa ferramenta permite uma verificação rápida do reconhecimento automático de entidades nomeadas e a correção de erros, levando em consideração as correções fornecidas pelo usuário, enquanto o modelo de Deep Learning aprende e desenvolve essas ações.

O trabalho também propõe dois modelos REN, sendo um deles baseado no framework spaCy e o outro na biblioteca Keras, com previsões complementares entre eles, com a finalidade de extrair entidades de boletins de ocorrência que descrevem uma ocorrência de homicídio. O presente trabalho compartilha pontos em comum com o estudo realizado por (ARAÚJO, 2019), uma vez que ambos propõem um modelo REN utilizando o framework spaCy.

3.0.5 Processamento de Linguagem Natural Aplicado a Reconhecimento de Entidades Nomeadas em Textos Legais em Português Brasileiro

No trabalho de (HECK, 2022), foi apresentado um modelo de Inteligência Artificial (IA) que se dedica à tarefa de REN em textos jurídicos escritos em português brasileiro. O modelo tem como objetivo reconhecer entidades como Pessoa, Tempo, Local, Organização, Legislação e Jurisprudência. Para isso, foram utilizadas as arquiteturas de redes neurais Bidirectional Long Short-Term Memory e Bidirectional Encoder Representation from Transformers, pré-treinada em língua portuguesa brasileira, com e sem camadas de Conditional Random Field. O conjunto de dados LeNER-Br foi utilizado para treinar os modelos. As métricas de avaliação do desempenho foram precisão, recall e F1-score, sendo que o modelo com melhor desempenho foi uma rede BERT-CRF, alcançando valores de 90,16% de precisão, 91,86% de recall e 91,00% de F1-score, superando o modelo baseline.

O presente trabalho se assemelha ao trabalho de (HECK, 2022) no uso do conjunto de dados LeNER-Br e no objetivo de reconhecer as entidades Pessoa, Tempo, Local, Organização, Legislação e Jurisprudência. No entanto, difere em relação às arquiteturas de redes neurais utilizadas.

3.1 Análise Comparativa

Na Tabela 2, é apresentada uma comparação entre os trabalhos relacionados e o atual, enfatizando as principais abordagens utilizadas ou desenvolvidas por cada estudo.

Tabela 1 – Análise Comparativa entre os trabalhos relacionados e a proposta deste trabalho.

Trabalho	Dataset	Rede neural
(ARAUJO, 2018)	LeNER-Br	LSTM-CRF
(MOTA et al., 2021)	LeNER-Br + petições iniciais	BiLSTM-CRF e CNN
(ZANUZ; RIGO, 2022)	LeNER-Br	BERT
(HECK, 2022)	LeNER-Br	BiLSTM e BERT + CRF
(ARAÚJO, 2019)	BOs	BiLSTM-CRF e CNN
Trabalho proposto	LeNER-Br	CNN

Fonte: Elaboração própria.

Essa tabela proporciona uma visão detalhada dos diversos aspectos abordados por cada estudo, tornando mais fácil a identificação das lacunas existentes e ressaltando as contribuições específicas da proposta em questão.

Ao analisar os trabalhos correlatos, torna-se evidente a importância da pesquisa em lidar com as falhas identificadas, como o desempenho dos dois modelos spaCy para a tarefa de REN, bem como a criação da interface do usuário. A proposta visa preencher essas brechas e contribuir para o progresso do REN.

4 Desenvolvimento

Este capítulo descreve de forma detalhada o processo de desenvolvimento (ANEXO A e B) do projeto.

O projeto foi implementado utilizando a linguagem de programação Python, devido ao seu alto desempenho, produtividade e legibilidade, sendo ideal para trabalhar com técnicas de ML e RNAs (FOUNDATION, 2023). Além disso, foram utilizadas outras bibliotecas do Python, como spaCy e Streamlit. Uma outra tecnologia empregada foi o Jupyter Notebook, uma aplicação web de código aberto que permite a criação e compartilhamento de documentos interativos conhecidos como notebooks. Esses notebooks combinam elementos de código, texto e recursos multimídia, possibilitando aos usuários a criação e compartilhamento de relatórios, visualizações de dados, análises exploratórias e implementação de modelos, entre outras funcionalidades (LOCAWEB, 2023).

4.1 Base de Dados

Para o presente estudo, utilizou-se o conjunto de dados LeNER-Br[1], o qual constitui uma coleção de documentos legais em língua portuguesa anotados manualmente para o REN. A composição desse conjunto de dados envolveu a coleta de 66 documentos judiciais provenientes de diversos Tribunais brasileiros, incluindo tribunais de instâncias superiores e estaduais, tais como o Supremo Tribunal Federal, o Superior Tribunal de Justiça, o Tribunal de Justiça de Minas Gerais e o Tribunal de Contas da União. Além disso, foram coletados quatro documentos de legislação, incluindo a Lei Maria da Penha, totalizando 70 documentos.

A base de dados de (ARAUJO, 2018) abrange seis classes de entidades, a saber: "ORGANIZACAO"para organizações, "PESSOA"para indivíduos, "TEMPO"para expressões temporais, "LOCAL"para localizações, "LEGISLACAO"para leis e "JURISPRUDÊNCIA"para decisões judiciais, a quantificação de palavras associadas a cada categoria de entidade é exibida na Tabela 3. A marcação das entidades segue o esquema IOB (RAMSHAW; MARCUS, 1999), em que "B-"indica o início de uma entidade nomeada, "I-"indica que um token está dentro de uma entidade nomeada e "O-"indica que um token não pertence a nenhuma entidade nomeada.

Para a criação desse conjunto de dados, 50 documentos foram selecionados aleatoriamente para o conjunto de treinamento, enquanto 10 documentos foram destinados a cada um dos conjuntos de desenvolvimento e teste, conforme apresentado na Tabela 2.

[1] https://github.com/peluz/lener-br

Tabela 2 – Contagem de sentenças, tokens e documentos para cada conjunto.

Conjunto	Documentos	Sentenças	Tokens
Conjunto de treinamento	50	7,827	229,277
Conjunto de desenvolvimento	10	1,176	41,166
Conjunto de teste	10	1,389	47,630

Fonte: (ARAUJO, 2018).

A distribuição do número de palavras nas entidades nomeadas de cada classe pode ser observada na Tabela 3. É evidente que o conjunto de dados apresenta um desequilíbrio, com as entidades Localização, Tempo e Jurisprudência possuindo o menor número de dados de treinamento, o que pode alterar os resultados do modelo.

Tabela 3 – Contagem de palavras da entidade nomeada para cada conjunto.

Entidade	Conjunto de treinamento	Conjunto de desenvolvimento	Conjunto de teste
Pessoa	4,612	894	735
Jurisprudência	3,967	743	660
Tempo	2,343	543	260
Localização	1,417	244	132
Legislação	13,039	2,609	2,669
Organização	6,671	1,608	1,367

Fonte: (ARAUJO, 2018).

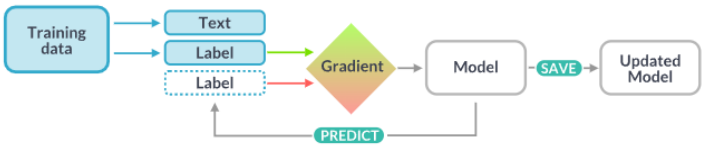
Os dados presentes nessa base estão disponíveis no formato CoNLL para REN, que é amplamente utilizado na conferência anual com o mesmo nome. Essa estrutura consiste em um arquivo de texto composto por duas colunas. Na primeira coluna, encontram-se as palavras e pontuações que compõem o texto, enquanto a segunda coluna contém as tags ou classes associadas a cada unidade linguística. Como parte do experimento, esses dados foram convertidos do formato CoNLL para um formato binário com a extensão .spacy. Essa extensão é usada como entrada para especificar um corpus de treinamento da biblioteca spaCy.

4.2 Treinamento dos modelos

Após preparar o conjunto de dados, esta etapa consiste em treinar os modelos para REN em textos do domínio jurídico. A Figura 7 mostra esse processo de treinamento dos modelos:

A biblioteca spaCy nos permite treinar modelos personalizados para cada finalidade usando dados adequadamente organizados para isso. O processo de treinamento dos modelos desse experimento consistiu primeiramente do carregamento de dois modelos

Figura 7 – Pipeline de treinamento dos modelos.



Fonte: (SPACY, 2023).

padrões da biblioteca (pt core news sm e pt core news lg), em seguida, os mesmos modelos foram alimentados com os dados de treinamento do dataset LeNER-Br que se constitui de textos de amostra e rótulos que se deseja prever.

Então os modelos são treinados por meio de um processo iterativo no qual as previsões dos modelos são comparadas com as anotações de referência para estimar o gradiente da perda, o gradiente da perda é então usado para calcular o gradiente dos pesos através da retropropagação, os gradientes indicam como os valores de peso devem ser alterados para que as previsões do modelo se tornem mais semelhantes aos rótulos de referência ao longo do tempo. Dessa forma os modelos estão treinados e ajustados para que possam ser salvos em disco e posteriormente carregados no futuro e usados para prever entidades nomeadas do domínio jurídico em textos em português (SPACY, 2023).

4.3 Aplicativo

O objetivo deste trabalho consiste em realizar uma comparação entre dois modelos de REN, com o intuito de resolver um problema de extração de informações a partir de textos. Para alcançar esse objetivo, foi desenvolvido um aplicativo web que possibilita a interação do usuário com os modelos e a visualização dos resultados obtidos. O desenvolvimento desse aplicativo web foi realizado utilizando o framework Streamlit, em conjunto com a plataforma Hugging Face.

O Streamlit é um framework em Python que facilita a criação de aplicativos web elegantes, destinados à utilização de modelos de machine learning ou visualização de dados. Por meio do Streamlit, é possível escrever código em Python e gerar interfaces gráficas contendo componentes como botões, sliders, gráficos e tabelas, entre outros. Além disso, o Streamlit oferece uma maneira simples de hospedar e compartilhar os aplicativos na web (AWARI, 2023).

A plataforma Hugging Face, por sua vez, oferece uma coleção de modelos pré-treinados para PLN e visão computacional, bem como ferramentas para carregar, treinar e avaliar novos modelos. O Hugging Face também disponibiliza um serviço chamado Spaces, que permite hospedar aplicativos web utilizando os modelos disponíveis na plataforma (FACE, 2023).

Para desenvolver o aplicativo web deste trabalho, foram realizados três passos principais, conforme apresentado na Figura 8:

Figura 8 – Fluxo de desenvolvimento do aplicativo.

Fonte: Elaboração própria.

1. Inicialmente, os dois modelos criados na seção 4.2 deste estudo foram carregados.
2. Em seguida, foi implementada uma interface gráfica utilizando o Streamlit. Essa interface permite que o usuário insira ou cole um texto, selecione um dos modelos de REN e visualize as entidades nomeadas reconhecidas pelo modelo, destacadas em cores diferentes.
3. Posteriormente, o aplicativo web foi hospedado no Spaces do Hugging Face. Essa plataforma possibilita a criação de um endereço online para acessar o aplicativo e compartilhá-lo com outras pessoas.

5 Resultados

Neste capítulo, são apresentados os resultados obtidos na execução do trabalho. Os experimentos realizados, considerando o conjunto de dados LeNER-Br e as duas arquiteturas, foram analisadas em relação às medidas de avaliação indicadas na Seção 2.4 deste trabalho com a ajuda da biblioteca spaCy e suas funções de avaliação de modelos.

5.1 Modelo small

O primeiro experimento realizado consistiu no treinamento do modelo pt_core_news_sm do framework spaCy com o conjunto de dados LeNER-Br. Esse modelo demonstrou resultados satisfatórios ao identificar informações como Legislação, Jurisprudência, Organização, pessoa, tempo e local.

Na Tabela 4 a seguir, são apresentados os valores das métricas precisão, recall e F-score para intervalos de caracteres de token. Além disso, a Tabela 5 exibe os resultados dessas mesmas métricas por tipo de entidade do modelo.

Tabela 4 – Resultados do modelo sm por token.

Modelo	Precisão	Recall	F1-score
pt_core_news_sm	82.73%	80.14%	81.42%

Fonte: Elaboração própria.

Tabela 5 – Resultados do modelo pt_core_news_sm por tipo de entidade.

Entidade	Precisão	Recall	F1-score
JURISPRUDENCIA	71.11%	69.19%	70.14%
ORGANIZACAO	77.31%	76.85%	77.08%
PESSOA	82.97%	81.55%	82.25%
TEMPO	94.97%	88.54%	91.64%
LOCAL	67.50%	57.45%	62.07%
LEGISLACAO	97.44%	87.57%	89.46%

Fonte: Elaboração própria.

Conforme observado na Tabela 4 e na Tabela 5, o modelo obteve um valor de precisão superior a 82%, considerado satisfatório. Com essa precisão, o modelo é capaz de extrair várias informações dos textos jurídicos de forma eficiente.

5.2 Modelo large

O próximo experimento realizado consistiu no treinamento do modelo pt_core_news_lg do framework spaCy com o mesmo conjunto de dados utilizado no primeiro experimento. Os resultados são apresentados nas Tabela 6 e Tabela 7, onde são exibidos os valores das métricas precisão, recall e F-score para intervalos de caracteres de token e os resultados dessas mesmas métricas por tipo de entidade do modelo, respectivamente.

Tabela 6 – Resultados do modelo lg por token.

Modelo	Precisão	Recall	F1-score
pt_core_news_lg	84.79%	82.75%	83.76%

Fonte: Elaboração própria.

Tabela 7 – Resultados do modelo pt_core_news_lg por tipo de entidade.

Entidade	Precisão	Recall	F1-score
JURISPRUDENCIA	82.00%	66.49%	73.43%
ORGANIZACAO	83.86%	79.84%	81.80%
PESSOA	82.57%	85.41%	83.97%
TEMPO	97.66%	86.98%	92.01%
LOCAL	57.58%	80.85%	67.26%
LEGISLACAO	87.31%	91.01%	89.12%

Fonte: Elaboração própria.

Como mencionado nas Tabela 6 e Tabela 7, o modelo alcançou uma taxa de acurácia acima de 84%, considerada aceitável. Com essa taxa, esse modelo igualmente tem a capacidade de extrair múltiplas informações dos documentos jurídicos de maneira eficaz.

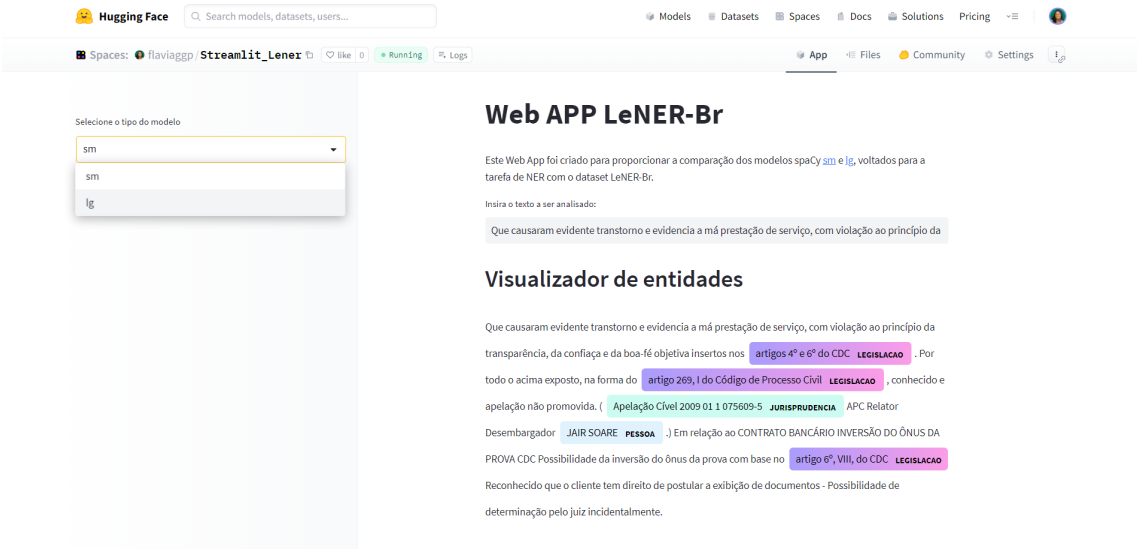
5.3 Aplicativo

O aplicativo está disponibilizado no Hugging Face Spaces[2]. Dessa forma, é possível que o usuário compare os resultados dos 2 modelos REN "sm"e "lg"ajustados no domínio jurídico brasileiro, produzidos neste trabalho.

O APP permite que o usuário teste ambos os modelos para a extração de entidades a partir de um texto. A tela inicial do aplicativo é ilustrada na Figura 9, onde é apresentado um exemplo de texto jurídico em uma área de texto. Os usuários têm a opção de realizar modificações ou inserir um novo texto. A extração das entidades é exibida por meio de marcações visuais no texto, em que cada entidade é associada a uma cor específica.

[2] https://huggingface.co/spaces/flaviaggp/Streamlit_Lener

Figura 9 – Interface do aplicativo



Fonte: Elaboração própria.

Além disso, é possível alternar entre os dois modelos por meio de uma caixa de seleção localizada à esquerda.

5.4 Comparação de resultados

A comparação entre os resultados obtidos neste projeto e os apresentados no modelo baseline, do trabalho de (ARAUJO, 2018), encontra-se na Tabela 8. É possível observar que, entre os modelos, o BiLSTM-CRF de (ARAUJO, 2018) obteve resultados superiores, com uma diferença de aproximadamente 3 pontos percentuais. No entanto, as Tabela 6, Tabela 7 e Tabela 9 demonstram que, quando se trata das métricas por entidade, os modelos desenvolvidos neste trabalho superam o modelo de (ARAUJO, 2018). O modelo "sm"apresentou melhor desempenho nas classes LEGISLAÇÃO e TEMPO, enquanto o modelo "lg"se destacou nas classes JURISPRUDÊNCIA e TEMPO. A entidade com pior desempenho em ambos os modelos foi LOCAL. Isso ocorre porque, segundo (ARAUJO, 2018), entidades dessa classe são as menos frequentes no conjunto de dados, representando 0,61% e 0,28% das palavras nos conjuntos de treinamento e teste, respectivamente. Isso dificulta o reconhecimento dessa entidade pelo modelo treinado, que tende a classificá-la erroneamente como PESSOA ou ORGANIZAÇÃO.

Tabela 8 – Comparação dos resultados dos modelos de REN.

Modelo	Precisão	Recall	F1-score
pt_core_news_sm	82.73%	80.14%	81.42%
pt_core_news_lg	84.79%	82.75%	83.76%
(ARAUJO, 2018)	87,98%	85,29%	86,61%

Fonte: Elaboração própria.

Tabela 9 – Resultados do modelo BiLSTM-CRF de (ARAUJO, 2018).

Entidade	Precisão	Recall	F1-score
JURISPRUDENCIA	79.29%	84.86%	81.98%
ORGANIZACAO	88.30%	82.83%	85.48%
PESSOA	85.58%	78.97%	82.14%
TEMPO	91.30%	87.50%	89.36%
LOCAL	69.77%	63.83%	66.67%
LEGISLACAO	93.93%	94.18%	94.06%

Fonte: (ARAUJO, 2018).

Em relação aos modelos desenvolvidos neste estudo, o modelo "lg"obteve melhor desempenho em comparação ao modelo "sm", de acordo com as métricas de avaliação apresentadas na Tabela 8, como esperado. Isso se deve ao fato de que o modelo "lg"é mais robusto, com 581 MB, possui um vocabulário mais abrangente, sintaxe, entidades e vetores dos tokens e classes gramaticais, sendo adequado para tarefas de alto nível que requerem alta precisão e complexidade.

Os resultados foram considerados satisfatórios, uma vez que ambos os modelos desenvolvidos durante este experimento obtiveram pontuações de F1-score acima de 80% e também superaram o baseline de (ARAUJO, 2018)no reconhecimento de algumas entidades.

6 Conclusão

Neste estudo, foram apresentados os experimentos e os resultados do projeto conduzido, onde técnicas de Inteligência Artificial e PLN foram empregadas para executar a tarefa de reconhecimento de entidade nomeada em textos legais escritos em língua portuguesa brasileira.

O objetivo geral de desenvolver um modelo treinado capaz de identificar citações de entidades pré-definidas em textos legais foi alcançado. Além disso, os objetivos específicos foram cumpridos por meio da pesquisa e utilização de dois modelos do spaCy, sendo eles, a comparação dos modelos desenvolvidos neste estudo e a criação de um aplicativo para interação dos usuários com os modelos.

Com base nos resultados apresentados no capítulo de resultados (Capítulo 5), é possível observar resultados satisfatórios nos modelos de REN utilizando o spaCy, com precisões de superiores a 80%. Isso evidencia que o framework spaCy é capaz de gerar modelos eficientes para o REN com poucas linhas de código e de forma ágil.

Para trabalhos futuros, é recomendado expandir o conjunto de dados de treinamento, incluindo mais documentos jurídicos e legislações. Além disso, é sugerido aumentar a proporção de palavras relacionadas à classe "LOCAL", que possui menor representação no conjunto de dados. Uma possível abordagem para melhorar o desempenho do modelo seria utilizar um modelo pré-treinado com textos legais, o que, aliado às demais sugestões mencionadas, tem potencial para alcançar um desempenho superior. Em relação ao APP é recomendado que em suas próximas versões o mesmo aceite aquivos .pdf e .docx, que são os formatos dos documentos reais do judiciário.

Referências

ANYOHA, R. The history of artificial intelligence. *Harvard University*, 2017. Citado na página 22.

ARAUJO, P. H. L. de. Lener-br: A dataset for named entity recognition in brazilian legal text. 2018. Disponível em: ⟨https://link.springer.com/chapter/10.1007/978-3-319-99722-3_32⟩. Citado 8 vezes nas páginas 13, 35, 36, 37, 38, 39, 44 e 45.

ARAÚJO, N. D. S. Reconhecimento de entidades nomeadas em textos de boletins de ocorrÊncias. 2019. Disponível em: ⟨https://repositorio.ufc.br/bitstream/riufc/49711/1/2019_tcc_nsaraujo.pdf⟩. Citado 2 vezes nas páginas 36 e 37.

AWARI. Streamlit: A biblioteca python que revoluciona o desenvolvimento de aplicações web. 2023. Disponível em: ⟨https://awari.com.br/streamlit-python/⟩. Citado na página 40.

BIRD, S.; KLEIN, E.; LOPER, E. *Natural Language Processing with Python*: Analyzing text with the natural language toolkit. 1. ed. [S.l.]: O'Reilly Media, 2009. Citado na página 29.

COELHO, L. Machine learning: O que é, conceito e definição. 2022. Disponível em: ⟨https://cetax.com.br/machine-learning/⟩. Acesso em: 26 fevereiro 2023. Citado na página 28.

DALE, R. Law and word order: Nlp in legal tech. *Natural Language Engineering*, v. 25, n. 1, p. 211–217, 2019. Disponível em: ⟨https://www.cambridge.org/core/journals/natural-language-engineering/article/law-and-word-order-nlp-in-legal-tech/E8CC6743F2FCCFD29FBC16A82F7F9B2A⟩. Citado na página 19.

DAS, S. Cnn architectures: Lenet, alexnet, vgg, googlenet, resnet and more.... 2017. Disponível em: ⟨https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df50⟩. Citado na página 32.

DATAPEAKER. O que são redes convolucionais: uma breve explicação. 2021. Disponível em: ⟨http://twixar.me/KHxm⟩. Citado na página 31.

FACE, H. Hugging face hub documentation. 2023. Disponível em: ⟨https://huggingface.co/docs/hub/index⟩. Citado na página 40.

FACELI, K. et al. *Inteligência Artificial*: Uma abordagem de aprendizado de máquina. 2nd. ed. [S.l.]: LTC, 2021. Citado na página 28.

FIGUEIREDO, F. A. P. de. Tp555 - inteligência artificial e machine learning: Introdução. Citado na página 31.

FOUNDATION, P. S. O tutorial de python. 2023. Disponível em: ⟨https://docs.python.org/pt-br/3/tutorial/index.html⟩. Citado na página 38.

GONCALVES, L. Reconhecimento de entidades nomeadas (ner) - o que é? quais são as aplicações? 2017. Disponível em: ⟨http://twixar.me/mHxm⟩. Citado na página 19.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. ⟨http://www.deeplearningbook.org⟩. Citado na página 32.

GUIMARAES, F. Rede neural convolucional (cnn) – o que é e como funciona. 2021. Disponível em: ⟨https://mundoprojetado.com.br/rede-neural-convolucional/⟩. Citado na página 31.

HAYKIN, S. *Redes Neurais- Princípios e Práticas*. 2. ed. [S.l.]: BOOKMAN, 2001. Citado na página 30.

HAYKIN, S. *Neural networks and learning machines*. 3. ed. [S.l.]: Upper Saddle River: Prentice Hall, 2009. Citado na página 30.

HECK, A. R. Processamento de linguagem natural aplicado a reconhecimento de entidades nomeadas em textos legais em português brasileiro. 2022. Citado 2 vezes nas páginas 33 e 37.

INSIGHTS, S. Machine learning - o que é e qual sua importância? 2021. Citado na página 28.

JURAFSKY, D.; MARTIN, J. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. [S.l.: s.n.], 2008. v. 2. Citado 2 vezes nas páginas 23 e 24.

JUSTIçA, C. nacional de. Justiça em números 2022. 2022. Disponível em: ⟨https://www.cnj.jus.br/wp-content/uploads/2022/09/justica-em-numeros-2022-1.pdf⟩. Citado na página 20.

KONKOL, M.; BRYCHCíN, T.; KONOPíK, M. Latent semantics in named entity recognition. *Expert Systems with Applications*, v. 42, n. 7, p. 3470–3479, 2015. ISSN 0957-4174. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S0957417414007933⟩. Citado na página 27.

LIMA, R. T. M. de O. Arquitetura baseada em redes neurais convolucionais para a detecÇÃo automÁtica da covid-19 usando imagens de raio x. 2021. Citado na página 30.

LOCAWEB. Jupyter notebook: saiba o que é e como utilizar a ferramenta. 2023. Disponível em: ⟨https://www.locaweb.com.br/blog/temas/codigo-aberto/jupyter-notebook-o-que-e/⟩. Citado na página 38.

MA, X.; XIA, F. Unsupervised dependency parsing with transferring distribution via parallel guidance and entropy regularization. In: . [S.l.: s.n.], 2014. v. 1. Citado na página 28.

MANDELLI, P. Redes neurais convolucionais: O que são e como aplicá-las. 2023. Disponível em: ⟨https://domineia.com/redes-neurais-convolucionais/⟩. Citado na página 31.

MARSHALL, C. What is named entity recognition (ner) and how can i use it? 2019. Disponível em: ⟨https://medium.com/mysuperai/what-is-named-entity-recognition-ner-andhow-can-i-use-it-2b68cf6f545d⟩. Citado na página 28.

MCCARTHY, J. What is artificial intelligence? *Computer Science Department Stanford University*, 2007. Citado na página 22.

MONARD, M. C.; BARANAUKAS, J. A. Aplicações de inteligência artificial: Uma visão geral. *Instituto de Ciencias Matematicas e de Computacao de Sao Carlos*, 2000. Citado na página 23.

MOTA, C. et al. Reconhecimento de entidades nomeadas em documentos jurídicos em português utilizando redes neurais. In: *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*. Porto Alegre, RS, Brasil: SBC, 2021. p. 130–140. ISSN 2763-9061. Disponível em: ⟨https://sol.sbc.org.br/index.php/eniac/article/view/18247⟩. Citado 4 vezes nas páginas 19, 35, 36 e 37.

NADEAU, D.; SEKINE, S. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, v. 30, 2007. Citado 2 vezes nas páginas 27 e 28.

RAMSHAW, L. A.; MARCUS, M. P. Text chunking using transformation-based learning. In: _____. *Natural Language Processing Using Very Large Corpora*. Dordrecht: Springer Netherlands, 1999. p. 157–176. ISBN 978-94-017-2390-9. Disponível em: ⟨https://doi.org/10.1007/978-94-017-2390-9_10⟩. Citado na página 38.

RAUBER, T. W. Redes neurais artificiais. 2005. Citado na página 29.

REINA, E. Elevado número de processos pendentes atrapalha andamento da justiça no país. 2022. Disponível em: ⟨https://www.conjur.com.br/2022-jun-17/elevado-numero-processos-pendentes-atrapalha-andamento-justica⟩. Citado na página 20.

ROBERTS, E. Natural language processing. *Stanford University*, 2009. Citado na página 23.

SHANMUGAMANI, R.; ARUMUGAM, R. *Hands-On Natural Language Processing with Python*: A practical guide to applying deep learning architectures to your nlp applications. [S.l.]: Packt Publishing, 2018. Citado na página 24.

SHIRALY, K. Building production-grade spacy text classification pipelines for business data. 2023. Disponível em: ⟨https://www.width.ai/post/spacy-text-classification⟩. Citado na página 27.

SILVA, B. C. D. da. O estudo lingüístico-computacional da linguagem. *Letras de Hoje*, v. 41, n. 2, p. 103–138, 2006. Citado 2 vezes nas páginas 23 e 24.

SILVA, R. V. M. e. Diversidade e unidade: A aventura linguística do português. 1988. Disponível em: ⟨https://edisciplinas.usp.br/pluginfile.php/4533665/mod_label/intro/MATTOSeSILVA_DiversidadeUnidadeAventuraLinguisticaDoPortugues.pdf⟩. Citado na página 19.

SPACY. Industrial-strength natural language processing. 2023. Disponível em: ⟨https://spacy.io/⟩. Citado 3 vezes nas páginas 26, 27 e 40.

SPECK, R.; NGOMO, A.-C. N. Ensemble learning for named entity recognition. In: MIKA, P. et al. (Ed.). *The Semantic Web – ISWC 2014*. Cham: Springer International Publishing, 2014. p. 519–534. ISBN 978-3-319-11964-9. Citado na página 28.

WIKIPEDIA. Rede neural convolucional. 2020. Disponível em: ⟨https://pt.wikipedia.org/wiki/Rede_neural_convolucional⟩. Citado na página 31.

ZANUZ, L.; RIGO, S. J. Fostering judiciary applications with new fine-tuned models for legal named entity recognition in portuguese. *Computational Processing of the Portuguese Language*, p. 219–229, 2022. Disponível em: ⟨https://link.springer.com/chapter/10.1007/978-3-030-98305-5_21⟩. Citado 2 vezes nas páginas 36 e 37.

Anexos

ANEXO A – Código dos modelos

```
# REN jurídico

# Drive:
from google.colab import drive
drive.mount('/content/drive')

# Dowload do modelo sm:
!python -m spacy download pt_core_news_sm

# Dowload do modelo lg:
!python -m spacy download pt_core_news_lg

# Importando as bibliotecas
import spacy
from spacy.tokens import DocBin
from spacy.scorer import Scorer

# Convertendo o dataset do formato .conll para o formato .spacy:
!python -m spacy convert /content/drive/MyDrive/TCC/Pratica/leNER-Br/
    test/test.conll  /content/drive/MyDrive/TCC/Pratica/Programac a o/
    Resultados
!python -m spacy convert /content/drive/MyDrive/TCC/Pratica/leNER-Br/dev
    /dev.conll /content/drive/MyDrive/TCC/Pratica/Programac a o/
    Resultados
!python -m spacy convert /content/drive/MyDrive/TCC/Pratica/leNER-Br/
    train/train.conll /content/drive/MyDrive/TCC/Pratica/Programac a o/
    Resultados

# Treinando o modelo sm:
!python -m spacy init fill-config /content/drive/MyDrive/TCC/Pratica/
    Programac a o/base_config/base_config_pt_efficiency.cfg config_ef.
    cfg
!python -m spacy train /content/drive/MyDrive/TCC/Pratica/Resultados/
    config_ef.cfg --output ./modelo2 --paths.train /content/drive/MyDrive
    /TCC/Pratica/Programac a o/Resultados/train.spacy --paths.dev /
    content/drive/MyDrive/TCC/Pratica/Programac a o/Resultados/dev.
    spacy

# Treinando o modelo lg:
!python -m spacy init fill-config /content/drive/MyDrive/TCC/Pratica/
    Programac a o/base_config/base_config_pt_accuracy.cfg config_ac.cfg
!python -m spacy train /content/drive/MyDrive/TCC/Pratica/Resultados/
    config_ac.cfg --output ./modelo1 --paths.train /content/drive/MyDrive
    /TCC/Pratica/Programac a o/Resultados/train.spacy --paths.dev /
```

```
    content/drive/MyDrive/TCC/Pratica/Programac a o/Resultados/dev.
    spacy
30
31 # Testando o modelo sm:
32 nlp_ner = spacy.load("/content/modelo2/model-best")
33
34 doc = nlp_ner("Todavia, entendo que extrair da aludida norma o sentido
    expresso na redação acima implica desconstruir o significado do texto
     constitucional, o que é absolutamente vedado ao intérprete. Nesse
    sentido, cito Dimitri Dimoulis:    (...) ao intérprete não é dado
    escolher significados que não estejam abarcados pela moldura da norma
    . Interpretar não pode significar violentar a norma.    (Positivismo
    Jurídico. São Paulo: Método, 2006, p. 220).59. Dessa forma, deve-se
    tomar o sentido etimológico como limite da atividade interpretativa,
    a qual não pode superado, a ponto de destruir a própria norma a ser
    interpretada. Ou, como diz Konrad Hesse,  o  texto da norma é o
    limite insuperável da atividade interpretativa.    (Elementos de
    Direito Constitucional da República Federal da Alemanha, Porto Alegre
    : Sergio Antonio Fabris, 2003, p. 71).")
35
36 colors = {"LEGISLACAO": "#F67DE3", "JURISPRUDENCIA": "#7DF6D9", "LOCAL":
    "#FFFFFF", "TEMPO":"#F0F8FF", "ORGANIZACAO":"#FAEBD7", "PESSOA":"#
    FFEFDB"}
37 options = {"colors": colors}
38
39 spacy.displacy.render(doc, style="ent", options= options, jupyter=True)
40
41 # Testando o modelo lg:
42 nlp_ner = spacy.load("/content/modelo1/model-best")
43
44 doc = nlp_ner("Todavia, entendo que extrair da aludida norma o sentido
    expresso na redação acima implica desconstruir o significado do texto
     constitucional, o que é absolutamente vedado ao intérprete. Nesse
    sentido, cito Dimitri Dimoulis:    (...) ao intérprete não é dado
    escolher significados que não estejam abarcados pela moldura da norma
    . Interpretar não pode significar violentar a norma.    (Positivismo
    Jurídico. São Paulo: Método, 2006, p. 220).59. Dessa forma, deve-se
    tomar o sentido etimológico como limite da atividade interpretativa,
    a qual não pode superado, a ponto de destruir a própria norma a ser
    interpretada. Ou, como diz Konrad Hesse,  o  texto da norma é o
    limite insuperável da atividade interpretativa.    (Elementos de
    Direito Constitucional da República Federal da Alemanha, Porto Alegre
    : Sergio Antonio Fabris, 2003, p. 71).")
45
46 colors = {"LEGISLACAO": "#F67DE3", "JURISPRUDENCIA": "#7DF6D9", "LOCAL":
    "#FFFFFF", "TEMPO":"#F0F8FF", "ORGANIZACAO":"#FAEBD7", "PESSOA":"#
    FFEFDB"}
```

```
47 options = {"colors": colors}
48
49 spacy.displacy.render(doc, style="ent", options= options, jupyter=True)
50
51 # Métricas de avaliação do modelo sm:
52 !python -m spacy evaluate /content/modelo2/model-best /content/drive/
      MyDrive/TCC/Pratica/Programac a o/Resultados/test.spacy --output /
      content/--gold-preproc
53
54 # Métricas de avaliação do modelo lg:
55 !python -m spacy evaluate /content/modelo1/model-best /content/drive/
      MyDrive/TCC/Pratica/Programac a o/Resultados/test.spacy --output /
      content/--gold-preproc
```

ANEXO B – Código do Web APP

```python
# Bibliotecas
import streamlit as st
import spacy
from spacy import displacy

#Título:
st.title("Web APP LeNER-Br")

#Descrição:
st.write("Este Web App foi criado para proporcionar a comparação dos modelos spaCy [sm](https://huggingface.co/flaviaggp/pt_pipeline) e [lg](https://huggingface.co/flaviaggp/pt_lg_pipeline), voltados para a tarefa de NER com o dataset LeNER-Br.")

#Texto:
input_text = st.text_input('Insira o texto a ser analisado:', 'Que causaram evidente transtorno e evidencia a má prestação de serviço, com violação ao princípio da transparência, da confiaça e da boa-fé objetiva insertos nos artigos 4 e 6 do CDC. Por todo o acima exposto, na forma do artigo 269, I do Código de Processo Civil, conhecido e apelação não promovida. (Apelação Cível 2009 01 1 075609-5 APC Relator Desembargador JAIR SOARE.) Em relação ao CONTRATO BANC RIO INVERS O DO NUS DA PROVA CDC Possibilidade da inversão do nus da prova com base no artigo 6 , VIII, do CDC Reconhecido que o cliente tem direito de postular a exibição de documentos - Possibilidade de determinação pelo juiz incidentalmente.')

# Função que carrega os modelos:
def load_models():
    sm_model = spacy.load("modelo_lener_sm")
    lg_model = spacy.load("modelo_lener_lg")
    models = {"sm": sm_model, "lg": lg_model}
    return models

models = load_models()
selected_type = st.sidebar.selectbox("Selecione o tipo do modelo", options=["sm", "lg"]) # caixa de seleção do modelo
selected_model = models[selected_type]
doc= selected_model(input_text) # função doc que processa o texto de acordo com a opção escolhida acima

# Cabeçalho
st.header("Visualizador de entidades")
```

```
29
30 #Cores:
31 colors = {"LEGISLACAO": "linear-gradient(90deg, #aa9cfc, #fc9ce7)", '
       JURISPRUDENCIA': "#ccfbf1", 'LOCAL': "#ffedd5", 'ORGANIZACAO': "#
       fae8ff", 'PESSOA': "#e0f2fe", 'TEMPO': "#fefde0", }
32 options = {"ents": ["LEGISLACAO", "JURISPRUDENCIA", "LOCAL", "
       ORGANIZACAO", "PESSOA", "TEMPO",], "colors": colors}
33
34 #Html:
35 ent_html = displacy.render(doc, style="ent", options=options, jupyter=
       False) # https://spacy.io/usage/visualizers
36
37 st.markdown(ent_html, unsafe_allow_html=True)
38
39 # streamlit run render.py
```