UNIVERSIDADE FEDERAL DO MARANHÃO

Engenharia da Computação

Clebson Mendonça Machado da Silva

Um Estudo com BERT para Reconhecimento de Entidades Nomeadas e Classificação de Peças Processuais no Domínio Jurídico Brasileiro

São Luís

2025

Clebson Mendonça Machado da Silva

Um Estudo com BERT para Reconhecimento de Entidades Nomeadas e Classificação de Peças Processuais no Domínio Jurídico Brasileiro

Trabalho de Conclusão de Curso apresentado ao curso de Engenharia da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Engenharia da Computação.

Orientador: Prof. Dr. Bruno Feres de Souza

São Luís

2025

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Diretoria Integrada de Bibliotecas/UFMA

Mendonça Machado da Silva, Clebson.
Um Estudo com BERT para Reconhecimento de Entidades Nomeadas e Classificação de Peças Processuais no Domínio Jurídico Brasileiro / Clebson Mendonça Machado da Silva. - 2025.
49 f.

Orientador(a): Bruno Feres de Souza.
Curso de Engenharia da Computação, Universidade Federal do Maranhão, São Luís, 2025.

1. Bert. 2. Processamento de Linguagem Natural No Domínio Jurídico. 3. Classificação de Textos Jurídicos. 4. Reconhecimento de Entidades Nomeadas. I. Feres de Souza, Bruno. II. Título.

Clebson Mendonça Machado da Silva

Um Estudo com BERT para Reconhecimento de Entidades Nomeadas e Classificação de Peças Processuais no Domínio Jurídico Brasileiro

Trabalho de Conclusão de Curso apresentado ao curso de Engenharia da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Engenharia da Computação.

Trabalho _____ em São Luís, 7 de agosto de 2025:

Prof. Dr. Bruno Feres de Souza
Orientador

Prof. Dr. Paulo Rogério de Almeira Ribeiro
Examinador

Profa. Dra. Sergio Souza Costa
Examinador

São Luís
2025

Agradecimentos

A Deus, por nos guiar para sermos bons samaritanos.

De forma eterna, ao meu pai, Geraldo Machado da Silva, que hoje ausente fisicamente, sempre esteve presente nessa caminhada.

Aos meus irmãos, meus amigos eternos, e a todos os meus familiares.

Aos meus amigos, tanto aos novos que surgiram nesta caminhada, quanto aos antigos que estiveram presentes quando precisei deles e nos momentos em que me fiz necessitado.

Ao orientador, professor Dr. Bruno Feres de Souza, cuja forma acolhedora de me receber como orientando é algo que levarei comigo para sempre.

Ao professor Dr. Paulo Rogério de Almeira Ribeiro, não apenas pelo excelente papel como professor e orientador durante um "pedaço" dessa jornada, mas por sua compreensão e forças nos momentos necessários e incentivos, terá sempre minha gratidão.

E a todos os professores do BICT e da Engenharia da Computação da UFMA, por tornarem essa jornada uma experiência única e marcante.

"Irmão! Não me abati e nem caí em desânimo.
A vida é vida em qualquer lugar,
a vida está em nós mesmos e não fora.
Ao meu lado haverá pessoas,
e ser homem entre elas e assim permanecer para sempre,
quaisquer que sejam os infortúnios,
sem perder a coragem nem cair em desânimo —
eis em que consiste a vida,
em que consiste o seu objetivo."

Dostoiévski

Resumo

O sistema judicial brasileiro enfrenta um desafio crítico de eficiência, encerrando o ano de 2023 com aproximadamente 83,8 milhões de processos pendentes. Diante desse cenário, as técnicas de Processamento de Linguagem Natural (PLN) surgem como uma solução estratégica para a automação de atividades e auxílio na tomada de decisão . Embora modelos de *machine learning* sejam frequentemente aplicados para essa automação, a natureza complexa e os jargões dos textos jurídicos exigem abordagens mais especializadas. A fim de criar um sistema especializado para esse domínio de texto, foi desenvolvido um pipeline de PLN que emprega modelos da arquitetura BERT para as tarefas de Classificação de Textos e Reconhecimento de Entidades Nomeadas (NER) utilizando a base de dados *Iudicium Textum Dataset*. Os resultados obtidos após o ajuste fino (*fine-tuning*) do modelo *BERTimbau* alcançaram para o Classificador de Texto uma acurácia de 98,31%, enquanto o modelo para o NER obteve um *F1-Score* de 87,64%. Com este desempenho, o sistema desenvolvido se apresenta como uma ferramenta poderosa para o avanço da PLN no setor jurídico brasileiro.

Palavras-chave: BERT; Processamento de Linguagem Natural no Domínio Jurídico; Classificação de Textos Jurídicos; Reconhecimento de Entidades Nomeadas.

Abstract

The Brazilian judicial system faces a critical efficiency challenge, having ended 2023 with approximately 83.8 million pending cases. In this scenario, Natural Language Processing (NLP) techniques emerge as a strategic solution for automating activities and assisting in decision-making. Although machine learning models are often applied for this automation, the complex nature and jargon of legal texts demand more specialized approaches. To create a specialized system for this text domain, an NLP pipeline was developed, employing models from the BERT architecture for the tasks of Text Classification and Named Entity Recognition (NER) using the *Iudicium Textum Dataset*. The results obtained after fine-tuning the *BERTimbau* model showed an accuracy of 98.31% for the Text Classifier, while the NER model achieved an *F1-Score* of 87.64%. With this performance, the developed system presents itself as a powerful tool for the advancement of NLP in the Brazilian legal sector.

Keywords: BERT; Legal Domain Natural Language Processing; Legal Text Classification; Named Entity Recognition.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

C3SL	*Centro de Computação Científica e Software Livre*
IA	*Inteligência Artificial*
ITD	*Iudicium Textum Dataset*
LFs	*Labeling Functions*
LSTM	*Long Short-Term Memory*
MLM	*Modelo de Linguagem Mascarada*
NER	*Reconhecimento de Entidades Nomeadas*
PLN	*Processamento de Linguagem Natural*
RNNs	*Redes Neurais Recorrentes*
STF	*Supremo Tribunal Federal*
UFMA	*Universidade Federal do Maranhão*
UFPR	*Universidade Federal do Paraná*

Sumário

1 Introdução

O avanço da litigiosidade no Brasil impõe um desafio crescente para a eficiência do Poder Judiciário. Dados do relatório "Justiça em Números - 2024"(Conselho Nacional de Justiça, 2024, p. 133) mostram que o sistema judicial encerrou o ano de 2023 com 83,8 milhões de processos pendentes. Desse total, 63,6 milhões encontravam-se em tramitação ativa e aguardavam uma solução definitiva por parte dos magistrados e servidores. Contudo, a complexidade desse problema não está apenas nos números, mas na natureza textual de cada processo, posto que cada um desses milhões de casos demanda a análise de um vasto conjunto de documentos, tarefa que, consequentemente, consome muito tempo e recursos, representando o principal entrave para a eficiência do judiciário.

A gravidade desse problema fica mais evidente quando se analisa a tendência de novas demandas no acervo, que atingiu em 2023 seu maior valor na série histórica. A consequência mais concreta desse volume é expressa no indicador "Tempo de Giro do Acervo", que conforme aponta Conselho Nacional de Justiça (2024, p. 138), mantida a atual produtividade seriam necessários aproximadamente 2 anos e 5 meses para concluir todo o estoque processual, isso em um cenário hipotético em que não haveria inclusão de novos processos no acervo.

Com efeito, esse tempo de espera pode estar ligado diretamente às cargas de trabalho geradas pela análise de documentos. Assim, cada etapa processual, desde a petição inicial até os diversos recursos utilizados em instâncias superiores exigem a leitura, a interpretação e a classificação de peças jurídicas complexas, configurando, assim, como o principal gargalo operacional que ocasiona a morosidade processual. Relatórios sobre o mercado jurídico global apontam que a revisão de documentos é uma das atividades que mais consomem tempo em escritórios de advocacia, gerando altos custos que os escritórios repassam aos clientes, o que, por sua vez, cria uma forte demanda por automação e eficiência (Thomson Reuters Institute, 2022).

Por conseguinte, a extração manual de informações essenciais, como as partes envolvidas, as leis citadas, as datas dos atos e o próprio tipo de petição, torna-se uma tarefa bastante repetitiva e que exige alta demanda cognitiva. Diante disso, a busca por soluções que otimizem tais análises torna-se necessária. Nesse contexto, a aplicação de técnicas de Inteligência Artificial (IA), como Processamento de Linguagem Natural (PLN), surge como uma abordagem promissora para a estruturação e análise eficiente do conteúdo jurídico.

O PLN é uma área da IA dedicada à criação de sistemas computacionais com capacidade de compreender, processar e responder à linguagem humana (DAS; DAS, 2024).

Assim, considerando que o contexto jurídico é em sua essência textual, o PLN se torna uma ponte tecnológica ideal para a automação da análise documental, sendo que, por meio de suas técnicas, torna-se possível superar os limites da análise manual, permitindo que tarefas de extração e classificação de informações sejam executadas com velocidade e consistência superiores à capacidade humana.

Ainda, é importante ressaltar o avanço significativo do PLN proporcionados pelos modelos *Transformers* nos últimos anos, principalmente após a introdução do artigo *"Attention Is All You Need"* (VASWANI et al., 2017), que diferentemente de outros modelos, os modelos *Transformers* possuem um mecanismo chamado *"Attention"*, o que lhe confere uma capacidade excepcional de compreender o contexto, uma vez que ele considera a relevância de cada palavra em sentenças mais longas e complexas. Tal característica acaba sendo valiosa para a interpretação de textos que possuem uma semântica complexa, como é o caso dos textos jurídicos. Entre os principais modelos desses avanços estão o *Generative Pre-trained Transformer* (GPT) e o *Bidirectional Encoder Representations from Transformers* (BERT), que estabeleceram um novo estado da arte na compreensão de linguagem natural, tornando-se a tecnologia de escolha perfeita para desafios textuais sofisticados.

Por fim, reforçando a necessidade e a atualidade deste trabalho, é válido ressaltar que o avanço da IA deixou de ser uma possibilidade no Poder Judiciário Brasileiro, se concretizando na aprovação do Ato Normativo 0000563-47.2025.2.00.0000 pelo CNJ, que detalhou um conjunto de regras para o uso e a auditabilidade de ferramentas de IA (Conselho Nacional de Justiça, 2025). Entre as diversas diretrizes mencionadas no ato, é válido destacar três: a ênfase na transparência, na proteção de dados e crucialmente na manutenção da decisão final sob responsabilidade humana. Portanto, o presente trabalho, ao propor um modelo de PLN para extrair e classificar informações de peças jurídicas, insere-se no exato contexto fomentado pelo CNJ, de criar mecanismos especializados que auxiliem no aumento da eficiência operacional sem substituir o papel do julgador, servindo como uma ferramenta para mitigar o gargalo da análise documental.

1.1 Objetivos

Este trabalho, alinhado às diretrizes do Conselho Nacional de Justiça (CNJ), tem como objetivo investigar a aplicação de técnicas de Processamento de Linguagem Natural (PLN), utilizando modelos baseados na arquitetura *Transformer* no contexto jurídico de língua portuguesa, visando capturar as terminologias especializadas e os contextos semânticos desse domínio.

1.1.1 Objetivo Geral

Avaliar a eficácia de modelos de PLN, especialmente baseados em *Transformers*, na classificação e extração de informações em documentos jurídicos, com foco na automação da triagem documental no âmbito do Poder Judiciário.

1.1.2 Objetivos Específicos

- Aplicar o processo de *fine-tuning* em modelos pré-treinados para a tarefa de classificação de texto, categorizando documentos jurídicos em classes pré-definidas, como habeas corpus e agravo regimental.
- Aplicar o processo de *fine-tuning* para a tarefa de Reconhecimento de Entidades Nomeadas (NER), identificando e rotulando automaticamente elementos como pessoas, organizações, leis, datas, tipos de documentos, crimes e locais.
- Avaliar o desempenho dos modelos em cada tarefa por meio de métricas quantitativas (precisão, *recall*, F1-score), considerando a sua aplicabilidade prática no contexto jurídico.

1.2 Trabalhos Relacionados

Em um estudo sobre a adaptação de modelos de linguagem para domínios especializados, Chalkidis et al. (2020) investigaram estratégias para aplicar o BERT ao campo jurídico. Em sua metodologia, foram empregadas três abordagens: o uso do BERT-BASE original, a continuação do pré-treinamento em corpora jurídicos (LEGAL-BERT-FP) e o pré-treinamento do zero com dados de domínio (LEGAL-BERT-SC). Para isso, os autores coletaram um corpus diversificado de 12 GB de textos jurídicos em inglês. Como resultado, o estudo lançou a família de modelos LEGAL-BERT e mostrou que as variantes desse modelo quase sempre superaram o desempenho do BERT-BASE ajustado.

O NER pode ser abordado em diferentes contextos. No trabalho de Zhong et al. (2023), por exemplo, a técnica foi aplicada para extrair automaticamente entidades médicas de textos sobre reabilitação em chinês, com objetivo de construir um sistema de apoio à decisão, utilizando o modelo gerado por uma parte do dataset para aumento de dados. Os resultados demonstram a eficácia dessa abordagem, com o novo modelo alcançando um *F1-Score* de 86,55%.

Batista et al. (2021) conduziram um estudo comparativo no dataset LeNER-Br entre o BERT e uma abordagem baseada em ChatGPT. Os resultados demonstraram a superioridade do BERT em todas as métricas de avaliação, que segundo os autores, essa diferença pode ser justificada pela natureza das arquiteturas, já que o GPT-3, base do

NER-ChatGPT, é um modelo de linguagem treinado para tarefas gerais e unidirecional, enquanto o BERT utiliza uma arquitetura de codificador bidirecional.

Silveira et al. (2023) introduziram o LegalBert-pt, um modelo de linguagem pré-treinado e especializado para o domínio jurídico em português brasileiro, sendo a principal referência no domínio no contexto nacional. O trabalho, que teve como corpus diversos textos jurídicos do CNJ, avaliou a eficácia do modelo em tarefas como reconhecimento de entidades nomeadas e classificação de texto e o disponibilizou como uma ferramenta de código aberto e personalizável. Os resultados desse modelo mostraram que supera consistentemente os modelos de linguagem de domínio geral, enfatizando a importância da especialização para se obter resultados eficazes no domínio jurídico.

Fora do contexto brasileiro, o trabalho de Darji, Mitrović e Granitzer (2023) focou na tarefa de NER do domínio jurídico alemã, utilizando um conjunto de dados contendo 750 decisões judiciais e anotadas em dois níveis de granularidade, uma com granularidade grossa (7 classes) e outra com granularidade fina (19 classes). A metodologia consistiu no ajuste fino de um modelo BERT específico para a língua alemã, comparando seu desempenho com um modelo BilSTM-CRF+ que representava o estado da arte anterior. Os resultados mostraram superioridade da abordagem com BERT que superou na maioria das 19 classes detalhadas, alcançando, por exemplo, um F1-score de 99,21% para a entidade Juiz.

Portanto, os trabalhos citados não apenas ilustram a evolução das técnicas de NER, como também consolidam a tese de que a especialização, seja ela no nível do idioma ou do domínio, é um fator fundamental para alcançar alta performance. Logo, o presente trabalho se insere neste contexto, utilizando um modelo especializado para o português do Brasil (*bertimbau*) para investigar a extração de um conjunto específico de entidades e a classificação de peças processuais, buscando contribuir para a validação e expansão dessas técnicas no cenário do direito brasileiro.

2 Fundamentação Teórica

Neste capítulo serão explorados os conceitos fundamentais para o desenvolvimento deste estudo, tais como: a Arquitetura *Transformer* e seu Mecanismo de Atenção; BERT, o estado da arte dos modelos *Transformers*; os Embeddings Semânticos, a metodologia de Supervisão Fraca, e a tarefa de Reconhecimento de Entidades Nomeadas (NER).

2.1 Arquitetura *Transformer*

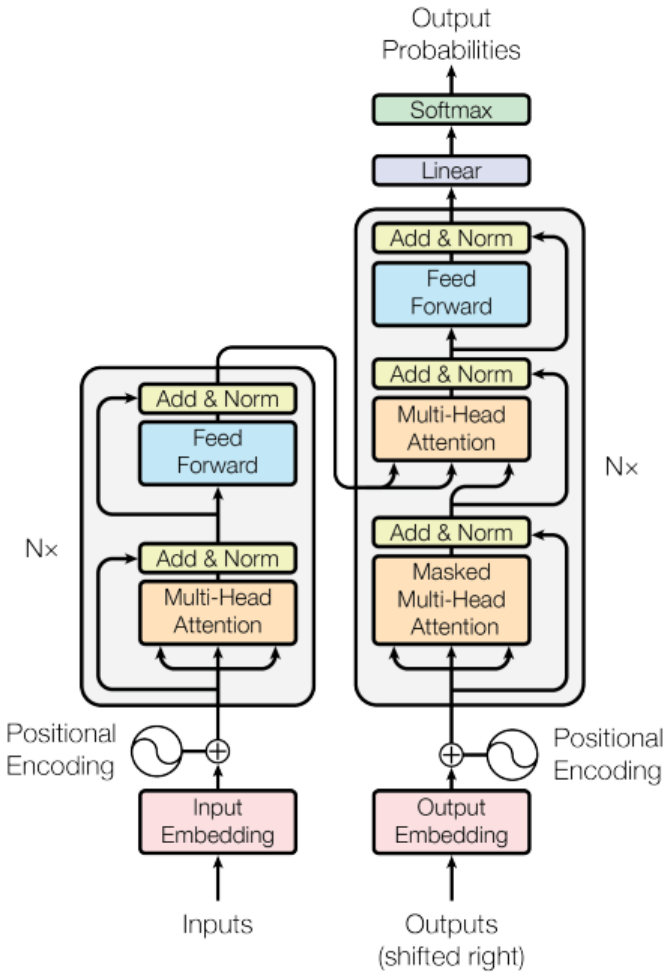
O advento dos modelos baseados na arquitetura *Transformers* representou um marco para o avanço do PLN. Apresentada por Vaswani et al. (2017), essa arquitetura rompeu com o paradigma sequencial das Redes Neurais Recorrentes (RNNs) e *Long Short-Term Memory* (LSTM) ao permitir que os modelos processassem sequências de texto em paralelo em vez de sequencialmente. Essa capacidade de paralelização está diretamente ligada ao seu mecanismo de atenção, em especial ao de autoatenção (*self-attention*), o qual permite que o modelo pondere a importância de cada palavra em uma sentença em relação a todas as outras, permitindo capturar dependências e relações contextuais complexas independentemente da distância entre elas no texto. Fundamentalmente, a arquitetura adota um design de codificador-decodificador (Figura 1) para realizar suas tarefas.

Detalhando a arquitetura, tanto o codificador quanto o decodificador são formados por um empilhamento de N camadas idênticas (Nx), embora a composição interna de cada camada seja distinta. Uma camada do codificador é composta por duas subcamadas: uma de *Multi-Head Attention* e uma rede neural *Feed-Forward*. Já a camada do decodificador contém três subcamadas: uma *Masked Multi-Head Attention*, uma segunda camada de *Encoder-Decoder Attention*, e por fim, a rede *Feed-Forward*. Para garantir a estabilidade em um modelo tão profundo, cada uma dessas subcamadas é seguida por uma conexão residual e uma camada de normalização (*Add & Norm*) [HE et al., 2016; BA; KIROS; HINTON, 2016].

2.1.1 O mecanismo de Atenção

O mecanismo de atenção é o núcleo central da arquitetura Transformer, na qual permite ao modelo ponderar a importância de diferentes partes da sequência de entrada. Uma função de atenção pode ser vista como um mapeamento que associa uma consulta (*query*) e um conjunto de pares chave-valor (*key-value*) a uma saída (*output*). No contexto do Transformer, todos esses elementos são representados como vetores. Assim, a saída é calculada como uma soma ponderada dos vetores de Valor, onde o peso de cada valor é

Figura 1 – Representação conceitual da arquitetura *Transformer*



Fonte: Vaswani et al. (2017).

determinado dinamicamente pela função de compatibilidade entre a Consulta e sua Chave correspondente.

A implementação específica de atenção utilizada no *Transformer* é denominada Atenção ao Produto Escalar (*Scaled Dot-Product Attention*), cuja operação é representada pela equação 2.1. A partir de uma matriz de entrada $X \in \mathbb{R}^{n \times d}$, na qual n é o comprimento da sequência e d é a dimensão dos *embeddings*, sendo as matrizes de consulta (Q), chave (K) e valor (V) geradas através de projeções lineares aprendidas:

- $\mathbf{Q} = XW_Q$: são as perguntas que cada palavra faz sobre as outras para identificar o que é relevante em seu contexto;
- $\mathbf{K} = XW_K$: serve como identificadores que capturam a essência de cada palavra, permitindo a correspondência com as perguntas das *queries*;
- $\mathbf{V} = XW_V$: contém os valores, ou seja, as informações de cada palavra que serão ponderadas e combinadas para formar a saída

Logo, a pontuação de atenção é então calculada pela Equação 2.1, em que o produto escalar entre a consulta (Q) e todas as chaves (K) é calculado. Para a estabilização dos gradientes e para evitar que o produtos escalares muito grandes saturem a função *softmax*, o resultado é dividido pelo fator de escala $\sqrt{d_k}$, sendo d_k a dimensão dos vetores de chave. Por fim, a função *softmax* é aplicada para obter os pesos de atenção, que são usados para ponderar os valores (V).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V \tag{2.1}$$

Atenção Multi-Cabeças (*Multi-Head Attention*)

A abordagem de Atenção Multi-Cabeças (*Multi-Head Attention*) permite que o modelo aprenda a focar em diferentes tipos de relações e subespaços de representação de forma simultânea. Cada *head* funciona como um especialista independente, analisando a sequência de entrada a partir de uma perspectiva única. No artigo original de Vaswani et al. (2017), por exemplo, a arquitetura emprega $h = 8$ *heads* em paralelo com uma dimensão $d_k = d_v = 64$. Ao final, os resultados dos *heads* são concatenados e que consolida as diversas perspectivas aprendidas, gerando uma representação final muito mais rica e robusta do que uma única camada de atenção conseguiria produzir.

Aplicação da Atenção no Modelo

O Transformer usa o mecanismo de Atenção de três maneiras ligeiramente diferentes dentro de sua arquitetura, cada uma com um propósito distinto para o fluxo de informação

- ***Encoder self-attention***: nesta camada, as matrizes de consulta (Q), chave (K) e valor (V) são todas derivadas da saída da camada anterior do próprio codificador. Isso permite que cada palavra na sequência de entrada preste atenção a todas as outras palavras na mesma sequência. O resultado é a criação de representações contextuais ricas, onde o vetor de cada palavra agora entende seu próprio papel e significado dentro da sentença completa;
- ***Decoder self-attention* (com máscara)**: de forma similar, esta camada permite que cada posição na sequência de saída preste atenção às posições anteriores na mesma sequência. A máscara é o ponto central aqui e ela é usada para impedir que uma posição preste atenção a posições subsequentes. Isso garante a propriedade autorregressiva do modelo, assegurando que a previsão de uma nova palavra dependa apenas da sequência de palavras já gerada, o que é fundamental para a coerência textual;

- ***Encoder-decoder attention***: esta é a conexão entre as duas metades da arquitetura. Nesta camada, os vetores de chave (K) e valor (V) vêm da saída final de toda a pilha de codificadores, enquanto os vetores de consulta (Q) vêm da subcamada de autoatenção mascarada do decodificador. Isso permite que o decodificador, a cada passo de geração, foque nas partes mais relevantes da sequência de entrada original para guiar a produção da próxima palavra na sequência de saída, alinhando efetivamente a saída ao contexto da entrada.

2.1.2 BERT - *Bidirectional Encoder Representations from Transformers*

O modelo BERT (*Bidirectional Encoder Representations from Transformers*), proposto por Devlin et al. (2019), representou um marco da área do PLN. Sua principal inovação frente aos outros foi a introdução de um pré-treinamento profundamente bidirecional, resolvendo uma limitação fundamental de modelos anteriores, como o GPT original, que operava de forma unidirecional, ou seja, processando o texto apenas da esquerda para a direita ou da direita para a esquerda. Tal limitação é particularmente prejudicial para tarefas de Compreensão de Linguagem Natural(*Natural Language Understanding*), pois o significado real de uma palavra frequentemente depende do contexto que a cerca por ambos os lados. Como resultado, o modelo se tornou a primeira abordagem baseada em ajuste fino a alcançar performance de ponta em um grande conjunto de aplicações, abrangendo e superando os resultados anteriores tanto em tarefas de nível de sentença quanto no de nível de token.

Representação de Entrada do BERT

Para que o BERT pudesse lidar com diferentes tipos de tarefa foi projetada uma estrutura de entrada flexível. O texto é primeiro processado por um *tokenizador* WordPiece que contém um vocabulário de 30.000 tokens. A entrada de modelo é então formatada com a adição de *tokens* especiais. O primeiro token de cada sequência é sempre o token [CLS], cuja representação final C é projetado para funcionar como um resumo de toda a sequência na saída do modelo. Já o *token* [SEP] é utilizado para separar diferentes sentenças. Por fim, o vetor de entrada para cada token é composto pela soma de três *embeddings* distintos:

- ***Token Embedding***: representa a palavra em si;
- ***Segment Embedding***: indica a qual sentença o token pertence, A ou B;
- ***Positional Embedding***: informa a posição do token na sequência.

Pré-Treinamento

Para alcançar a bidirecionalidade, o pré-treinamento do BERT propõe a execução conjunta de duas tarefas não supervisionadas: o Modelo de Linguagem Mascarada (*Masked Language Model*) e a Previsão de Próxima Sentença (*Next Setence Prediction*).

Enquanto os modelos unidirecionais focavam em prever a próxima palavra, a abordagem no Modelo de Linguagem Mascarada (MLM) é diferente. Nela, 15% das palavras da sequência são escolhidas aleatoriamente para a tarefa de previsão. Desses 15%, 80% são substituídos pelo token MASK, 10% por uma palavra aleatória e 10% mantêm a palavra original. Assim, o modelo aprende a preencher lacunas, a identificar e corrigir erros, além de gerar bons vetores para palavras reais, o que se torna particularmente importante nessa etapa de *fine-tuning*, uma vez que não haverá máscara nessa etapa (DEVLIN et al., 2019).

Para Previsão de Próxima Sentença, Devlin et al. (2019) mostra que essa tarefa foi criada devido à necessidade de que o BERT aprendesse a entender a relação entre as sentenças, o que seria útil para tarefas como as de perguntas e respostas. Esse processo é binário, então para cada par de sentenças "A"e "B"criados como exemplos, há duas possibilidades. Primeiro, 50% dos casos é rotulado com *IsNext*, o que representa que a sentença B de fato possui relação com a sentença A. Por outro lado, 50% dos casos são rotulados como *NotNext*, o que representa que B é uma sentença completamente aleatória retirada de outra parte do texto com nenhuma relação com a sentença A.

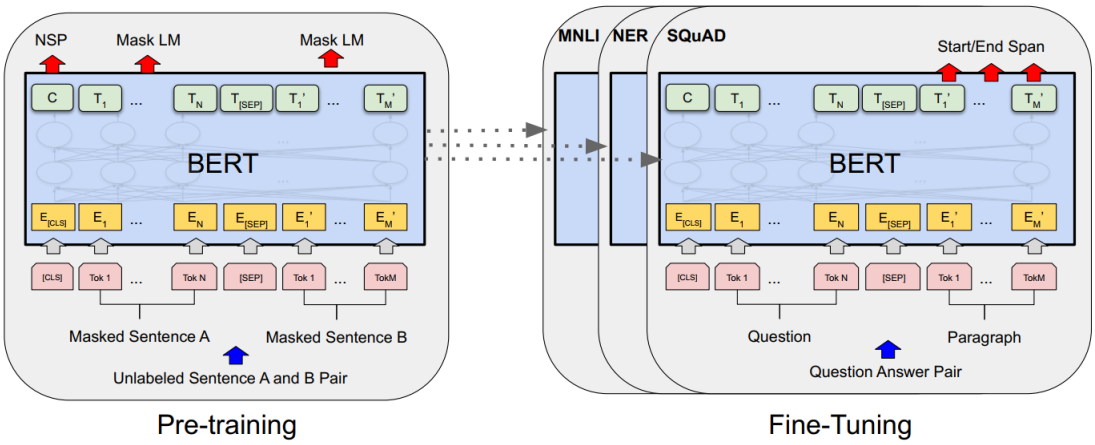
Ajuste Fino

Depois do pré-processamento, na qual o modelo adquire conhecimento profundo sobre a estrutura da linguagem, a segunda etapa fundamental é o ajuste fino (*fine-tuning*). Nessa fase, o modelo BERT pré-treinado é adaptado para uma tarefa supervisionada específica, como classificação ou reconhecimento de entidades. Para isso, uma pequena camada de saída, na qual é projetada para a tarefa em questão é adicionada ao topo da arquitetura. Assim, o modelo é treinado por um número reduzidos de época utilizando um conjunto de dados rotulados muito menor que o corpus de pré-treinamento. Essa é uma abordagem bastante eficiente, uma vez que ocorre a transferência de conhecimento linguístico generalista para o problema específico, evitando assim a necessidade de treinar uma rede neural complexa do zero.

Por fim, a simplicidade do processo de ajuste fino do BERT está situada em sua capacidade de modelar diferentes tipos de tarefas apenas adaptando as entradas e saídas. Conforme descrito por Devlin et al. (2019), a mesma arquitetura lida com diferentes problemas de forma unificada. Logo, para tarefas de nível de sentença, como a classificação de documentos realizada neste trabalho, a representação final do *token* especial [CLS] é

fornecida à camada de classificação para gerar um rótulo para o documento inteiro. Já para tarefas de nível de *token*, como o reconhecimento de entidades nomeadas (NER), a representação final de cada *token* individual da sequência é alimentada a uma camada de saída que atribui uma etiqueta especial a cada palavra. A dinâmica completa deste processo de duas fases, pré-treinamento seguido de ajuste fino, é ilustrada na Figura 2

Figura 2 – Procedimentos gerais de pré-treinamento e ajuste fino para BERT *Transformer*



Fonte: Devlin et al. (2019).

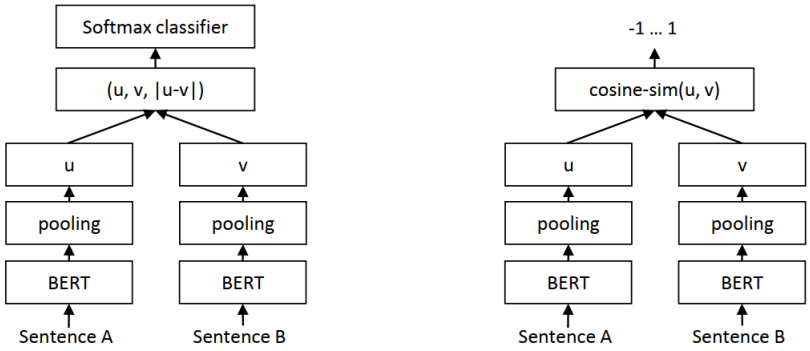
2.1.3 *Sentence-BERT*: *Embeddings* Semânticos para Análise de Similaridade

Embora a arquitetura BERT seja um motor *Transformer* poderoso, sua aplicação em tarefas de comparação de pares de sentença o faz funcionar como um codificador cruzado (*cross-encoder*). Isso requer que ambas as sentenças sejam processadas conjuntamente em uma única passagem pela rede para que a relação entre elas seja avaliada. Mas, conforme destacado por Reimers e Gurevych (2019), essa abordagem, apesar de ser precisa, torna-se computacionalmente inviável para tarefas de busca em larga escala. O gargalo reside na explosão combinatória de pares, assim em um conjunto de dados de $n = 10.000$ sentenças, por exemplo, seriam necessárias aproximadamente 50 milhões de inferências computacionais para encontrar o par mais similar, um processo que consumiria dezenas de horas em uma GPU moderna.

Para solucionar esse gargalo computacional do cruzado, os autores desenvolveram o Sentence-BERT (SBERT), uma arquitetura que adapta o BERT para que ele funcione como um bi-encoder. A principal modificação consistiu em adicionar uma camada de pooling na saída do modelo BERT, cuja função é agregar os vetores de todos os tokens em um único embedding de tamanho fixo que represente a sentença inteira. Foram utilizadas três diferentes estratégias para essa etapa, como o uso do vetor [CLS] ou a média dos vetores de saída, sendo a média (*MEAN-pooling*) a configuração que apresentou os melhores resultados.

Com essa arquitetura, o SBERT gera um embedding para cada sentença de forma independente. Uma vez que esses vetores são calculados e armazenados, a similaridade semântica entre quaisquer duas sentenças pode ser determinada de forma quase instantânea através da similaridade de cosseno, eliminando a explosão combinatória e viabilizando assim a aplicação em larga escala. A eficiência dessa abordagem foi validada experimentalmente, e conforme demonstrada por Reimers e Gurevych (2019), o modelo SBERT-NLI-large alcançou uma performance média de 76.55% em tarefas de similaridade semântica, enquanto a abordagem que usa o vetor [CLS] de um BERT padrão obteve um resultado médio de apenas 29.19%, mostrando a eficácia do SBERT em gerar *embeddings* de sentença de alta qualidade. A arquitetura do SBERT é ilustrada na Figura 3.

Figura 3 – Arquitetura SBERT. A representação da esquerda mostra a arquitetura SBERT com função objetivo de classificação. A representação da direita para a inferência.



Fonte: Reimers e Gurevych (2019).

2.1.4 Supervisão Fraca e a Geração Programática de Dados

A eficácia de modelos de aprendizado de máquina supervisionado está profundamente ligada à disponibilidade e à qualidade de grandes conjuntos de dados com anotações. Contudo, o processo de rotulagem manual representa um conhecido gargalo, sendo uma etapa que requer muito tempo e, às vezes, altos investimentos. No trabalho de Hendrycks et al. (2021), por exemplo, é destacado que a criação do dataset jurídico CUAD envolveu dezenas de especialistas para gerar mais de 13.000 anotações.

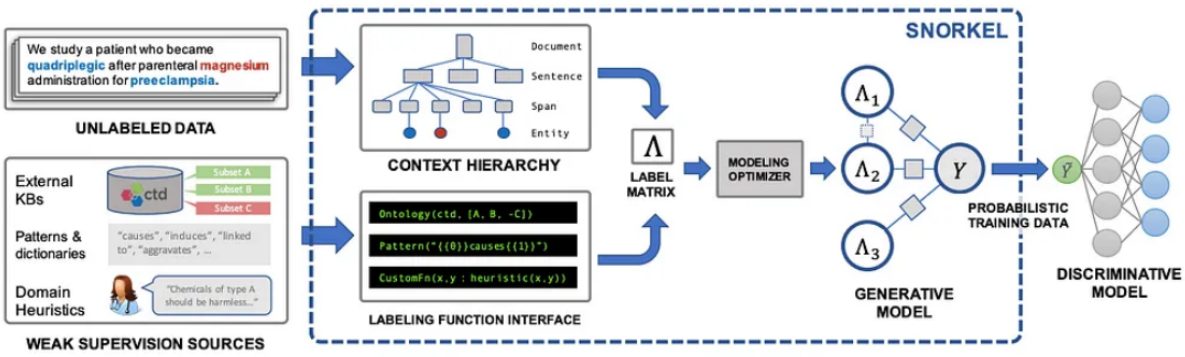
Ainda, Sun, Xu e Suominen (2021), em um estudo no domínio clínico, observaram que o acréscimo de anotações manuais detalhadas não resultou em uma melhoria significativa no desempenho do modelo, questionando assim o retorno sobre o investimento de tais processos. Neste contexto de alto custo e retorno incerto da rotulagem manual, as metodologias como a Supervisão Fraca (*Weak Supervision*) surgem como uma alternativa estratégica e eficiente.

A metodologia da Supervisão Fraca foi formalizada no projeto *Snorkel* por Ratner et al. (2017). O *Snorkel* é um sistema que permite a criação de dados de treinamento de forma programática, utilizando regras e heurísticas (as *Labeling Functions* - LFs) para gerar rótulos de forma automática ao invés de manualmente. Segundo os autores, com a utilização deste framework, especialistas construíram modelos 2,8 vezes mais rápido, além de terem aumentado o desempenho preditivo em uma média de 45,5% em comparação com sete horas de rotulagem manual.

Arquitetura do *Snorkel*, conforme ilustrado na Figura, 4 pode ser dividida em três etapas:

- ***Writing Labeling Functions***: a primeira etapa consiste em o especialista de domínio expressar seu conhecimento em forma de código, criando regras e heurísticas que rotulam os dados de forma programática.
- ***Modeling Accuracies and Correlations***: na segunda etapa, o *framework Snorkel* utiliza um modelo generativo para aprender as precisões e correlações de cada LFs criada anteriormente, analisando seus padrões de concordância e conflito para gerar rótulos probabilísticos.
- ***Training a Discriminative Model***: última etapa, o dataset com os rótulos probabilísticos é utilizado para treinar um modelo discriminativo final (como o BERT), cujo objetivo é aprender os padrões subjacentes dos dados e generalizar para além das heurísticas iniciais, superando seu desempenho.

Figura 4 – Visão geral do sistema *Snorkel* para criação de dados de treino com Supervisão Fraca



Fonte: Ratner et al. (2017).

Portanto, a metodologia *Snorkel* oferece um framework para superar o gargalo da rotulagem manual e sua relevância para o presente trabalho reside na aplicação do seu conceito central na fase de criação do dataset, na qual uma função de rotulagem baseada

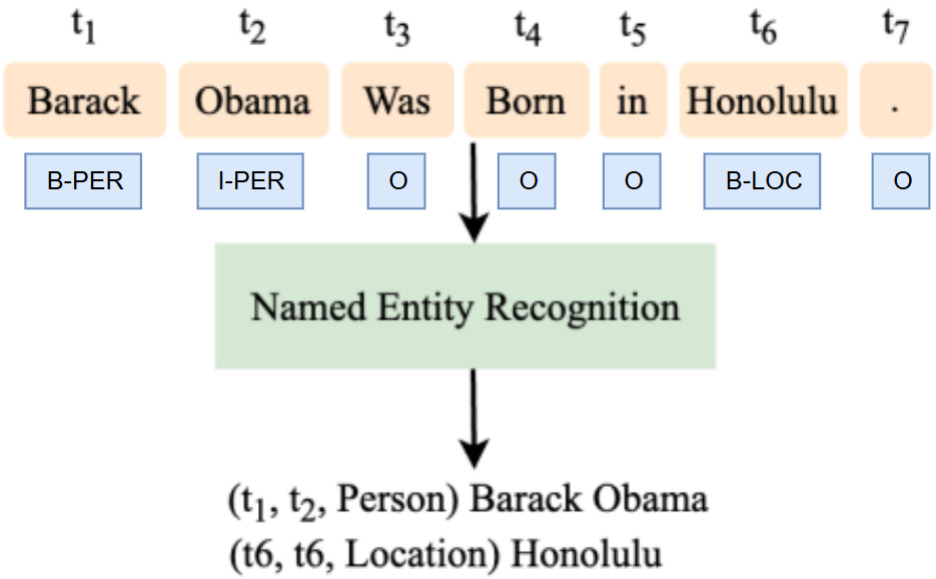
em similaridade semântica foi utilizada para gerar os rótulos de forma automática. Os detalhes e a implementação desta abordagem serão aprofundados no Capítulo 3.

2.1.5 Reconhecimento de Entidades Nomeadas (NER)

Reconhecimento de Entidades Nomeadas é uma tarefa fundamental da área de extração de informação em PLN, na qual consiste na identificação e classificação de entidades nomeadas dentro de textos não estruturados, como nomes de pessoas, locais e organizações (GRISHMAN; SUNDHEIM, 1996). Assim, o objetivo de um sistema de NER é, portanto, processar uma sequência de texto e extrair as ocorrências das entidades de interesse, atribuindo a cada uma sua respectiva categoria semântica.

Esse processo fica mais claro observando a Figura 5. Para uma sentença de entrada como *"Barack Obama was born in Honolulu"*, o modelo identifica que o trecho "Barack Obama"representa uma entidade do tipo PESSOA, e que "Honolulu"corresponde a uma do tipo LOCAL. Por fim, a saída do modelo consiste em um conjunto de tuplas que irá formalizar essa extração.

Figura 5 – Limites das entidades nomeadas gerados pelo NER juntamente com suas categorias associadas e representação dos respectivos rótulos para cada palavra dentro da sentença.



Fonte: Adaptado de Keraghel, Morbieu e Nadif (2024a).

Para que um modelo de aprendizado de máquina realize essa tarefa, a abordagem mais comum é tratá-la como uma tarefa de rotulação de sequência (*Sequence Labeling*). Nessa abordagem, cada token de uma sentença recebe um rótulo que indica se ele pertence a uma entidade nomeada e qual sua posição dentro dela. Um dos esquemas de rotulagem mais utilizados é o BIO (*Beginning*, *Inside*, *Outside*), proposto por Ramshaw e Marcus (1995), no qual cada *token* é rotulado como B-rótulo se for o início de uma entidade,

I-rótulo caso esteja dentro da mesma entidade, mas não seja o primeiro *token*, ou "O", caso não pertença a nenhuma entidade de interesse. O processo de rotulação é apresentado na Figura 5.

Atualmente, a abordagem estado da arte para a maioria das tarefas de NER é o ajuste fino de modelos de linguagem pré-treinados baseados na arquitetura *Transformer*. Estudos comparativos demonstram que esses modelos superam arquiteturas anteriores, como as redes BiLSTM-CRF, sendo que o sucesso de tais modelos está em sua capacidade de gerar representações vetoriais ricas em contexto, habilidade desenvolvida durante a fase de pré-treinamento. Esse profundo entendimento do contexto bidirecional permite que o modelo desambigue entidades de forma muito mais eficaz (KERAGHEL; MORBIEU; NADIF, 2024b).

3 Metodologia

A metodologia deste trabalho foi estruturada em cinco partes. A primeira parte está relacionada à *pipeline* experimental, na qual são definidos a base e os procedimentos de exploração dos dados, além do pré-processamento geral. A segunda etapa é dedicada à Classificação de Texto enquanto a terceira descreve os experimentos para o Reconhecimento de Entidades Nomeadas. Por fim, é descrito a configuração do ajuste fino e a avaliação na quarta e quinta etapa, respectivamente.

3.1 *Pipeline* Experimental

3.1.1 Definição da Base de Dados

A base utilizada neste trabalho foi a *Iudicium Textum Dataset* (ITD)[1]„ um conjunto de textos jurídicos desenvolvido e disponibilizado publicamente pelo Centro de Computação Científica e Software Livre (C3SL) da Universidade Federal do Paraná (UFPR). Essa base foi selecionada por ter um conjunto robusto de recursos textuais, por ser aberta e focada em documentos do Supremo Tribunal Federal (STF).

O corpus é formado por 41.353 acórdãos do STF que foram publicados entre os anos de 2010 e 2018. Esse acórdão é um documento de estrutura complexa, mas bem definida e resultante de um processo julgado pelo STF. A ITD preserva essas estruturas complexas, separando o documento original em seções constituintes, como a Ementa, que é o resumo da matéria, o Relatório, que é a descrição dos fatos, os Votos individuais de cada ministro e o Acordão, que é o resultado da votação (SOUSA; FABRO, 2019). As estatísticas detalhadas dos principais componentes da base podem ser encontradas na Tabela 1.

Tabela 1 – Resumo do Iudicium Textum Dataset (ITD) dos principais componente textuais.

Componente	**Documentos**	**Sentênças**	**Tokens**
Acórdãos	41.353	165.718	3.467.564
Relatórios	41.353	1.409.406	25.713.318
Extratoata	41.353	530.528	5.408.723
Total	**97.603**		

Fonte: autor.

Para os experimentos deste trabalho foram selecionados exclusivamente os textos dos relatórios da base ITD. Essa escolha se justifica pelo fato dos dados do relatório ser

[1] Disponível em: <https://dadosabertos.c3sl.ufpr.br/acordaos/json/>

a mais descritiva do acórdão, pois detalham os fatos e as circunstâncias do caso além da variedade de entidades encontradas no texto que servirão para as tarefas de NER e classificação, como menções a leis, locais, organizações e pessoas envolvidas.

3.1.2 Pré-Processamento dos Dados

Para a execução dos experimentos foi utilizado o arquivo DocumentosAcordaos.json da base ITD e convertido em uma estrutura tabular utilizando a biblioteca pandas. Uma análise exploratória do campo relatorio.texto revelou uma grande variação no comprimento dos documentos, com uma média de 3.337 caracteres e um valor máximo superior a 218 mil caracteres. Ainda, foi identificado que o conteúdo textual do documento era precedido por um preâmbulo com informações de cabeçalho, como o nome do ministro relator.

Essa informação já estava disponível em um campo estruturado e não fazia parte das descrições dos fatos, logo foi considerada uma potencial fonte de ruído para o modelo. Ainda, esse preâmbulo consumia um espaço valioso na janela de entrada da arquitetura *Transformer*, dado que modelos como o BERT operam com um limite de 512 tokens (DEVLIN et al., 2019), a presença desse texto aumentava o risco de que informações juridicamente relevantes, as que estão localizadas no final do documento, fossem perdidas pelo processo de truncamento que acontece na etapa de treinamento. Portanto, para minimizar esse problema foi removido sistematicamente o preâmbulo em todos os textos do *dataset* além de ter sido feita uma higienização nos dados, como remoção de múltiplos espaços, *tags html* e entre outros. O resultado desse processo é ilustrado na Tabela 2.

Tabela 2 – Processo de higienização dos campos de dados

Campo	**Texto Original**	**Texto Ajustado**
`texto`	'O Senhor Ministro Luís Roberto Barroso (Relator): 1. Trata-se de agravo regimental interposto...	'1. Trata-se de agravo regimental interposto...'

3.2 Classificação de Peças Jurídicas

Esta seção apresenta a metodologia empregada para construção e validação do classificador de peças processuais, na qual foi utilizado o corpus total de documento previamente descrito na Tabela 1.

3.2.1 Criação dos Rótulos

Dado que o corpus original não possui previamente o tipo de cada documento separado em uma coluna, uma etapa de supervisão fraca foi implementada. O desafio dessa

etapa foi criar um gabarito confiável, considerando a pluralidade de elementos encontrados no texto, para isso foram desenvolvidas duas abordagens.

A primeira abordagem foi puramente semântica, na qual se utilizou o modelo *paraphrase-multilingual-MiniLM-L12-v2* da biblioteca *sentence-transformers* por ser compatível com o idioma português, bem como ser otimizado para tarefas de similaridade entre sentenças. Neste método foram criados protótipos textuais que representam o significado ideal de cada categoria de peça processual. Tais protótipos foram desenvolvidos com base na exploração do dataset durante as marcações do NER e refinados com auxílio de dois profissionais na área jurídica, conforme detalhado abaixo. A classificação de cada documento foi então determinada pela maior similaridade de cossenos, calculada via biblioteca *scikit-learn* entre o vetor do documento e os vetores dos protótipos.

- **Ações / Remédios Constitucionais**
  - **Habeas Corpus**: petição inicial de habeas corpus com pedido de liminar para proteger a liberdade de locomoção contra prisão ilegal ou abuso de poder, visando a expedição de alvará de soltura ou o trancamento da ação penal.
  - **Mandado de Segurança**: Petição inicial de mandado de segurança para proteger direito líquido e certo contra ato ilegal de autoridade pública, não amparado por habeas corpus;
  - **Ação Direta de Inconstitucionalidade**: petição inicial de ação de controle concentrado para declarar a inconstitucionalidade de lei ou ato normativo federal ou estadual perante a Constituição;
  - **Reclamação**: petição inicial de reclamação constitucional para preservar a competência do tribunal ou garantir a autoridade de suas decisões e súmulas vinculantes.
- **Decisões Judiciais**
  - **Sentença**: ato decisório proferido por juiz de primeira instância que resolve o mérito e encerra a fase de conhecimento do processo;
  - **Decisão Monocrática**: decisão proferida por um único ministro ou relator em tribunal superior. Geralmente é o alvo de um agravo regimental;
  - **Acórdão**: julgamento colegiado de um tribunal que contém estrutura formal com ementa, relatório, voto do relator e a expressão final acordam, julgando um recurso ou ação.
- **Recursos**
  - **Apelação**: recurso interposto para contestar e buscar a reforma de uma sentença de primeira instância;

- **Agravo de Instrumento**: recurso contra decisão interlocutória, frequentemente para contestar a inadmissão de recurso especial ou extraordinário na origem;
- **Agravo Regimental**: recurso ou petição de agravo interno interposto contra decisão monocrática de relator, contendo as razões para submeter a matéria ao colegiado;
- **Embargos de Declaração**: recurso oposto contra uma decisão judicial para sanar omissão, contradição ou obscuridade;
- **Recurso Especial**: recurso ao STJ que alega violação ou interpretação divergente de lei federal.
- **Recurso Extraordinário**: recurso ao STF que alega violação direta da Constituição Federal e demonstra repercussão geral;
- **Recurso Ordinário em HC**: recurso interposto contra acórdão de tribunal superior que denegou uma ordem de habeas corpus.

A segunda abordagem implementada foi um sistema híbrido, combinando regras heurísticas com o método semântico. Inicialmente foi desenvolvida uma função em python para classificar a maior parte do conjunto de dados com base em palavras-chaves e padrões textuais. Como o texto possuía muita ambiguidade, ou seja, ocorrência de vários tipos de peças no mesmo documento, adotou-se uma heurística de primeira ocorrência, que determina o tipo de texto com base no primeiro termo jurídico encontrado no texto, como demonstrado na Tabela 2. Com essa abordagem, 871 documentos (2%) permaneceram "Não Identificados", sendo submetidos ao método semântico para lhes atribuir um rótulo.

3.3 Reconhecimento de Entidades Nomeadas - NER

Para o NER não foi utilizada o corpus total devido à robustez da base, como mostrado na Tabela 1. Para essa etapa foi selecionada uma amostra de dois mil documentos para o processo de anotação manual.

3.3.1 Anotação dos Dados

Inicialmente foram definidos oito classes semânticas. No entanto, a classe Jurisprudência foi removida devido as limitações de conhecimento para limitar o seu contexto, mas é válido que tal informação está presente no corpus. Assim, no escopo final foram definidas sete classes, sendo quatro no domínio jurídico (legislação, crime, data e documento) e sendo as três canônicas da literatura (pessoa, organização e local) como definido no trabalho de Grishman e Sundheim (1996).

Para o processo de anotação foi empregada a ferramenta de marcação de texto Docanno, uma plataforma especializada em tarefas de PLN. A funcionalidade utilizada

neste trabalho foi a de Rotulagem de Sequência (*Sequence Labeling)*. O resultado do processo resultou em um conjunto de dados robustos, totalizando um total de 65462 rótulos. A distribuição de entidades anotadas é mostrada na Tabela 3, apresentando a frequência de cada classe.

Tabela 3 – Distribuição e Frequência das Classes Anotadas no Corpus NER

Classe	**Ocorrências**	**Frequência (%)**
DOCUMENTO	23.842	36,4%
LEGISLACAO	15.880	24,3%
ORGANIZACAO	12.284	18,8%
DATA	5.629	8,6%
PESSOA	5.205	8,0%
CRIME	2.075	3,2%
LOCAL	547	0,8%
Total de Anotações	**65.462**	**100,0%**

A seguir, cada uma dessas classes é definida e detalhada com sua respectiva definição e um exemplo prático de anotação.

Classe Pessoa

Essa classe está relacionada com todos os envolvidos no processo judicial ou que são mencionados no documento. É válido ressaltar que, além do primeiro nome, foram marcados os sobrenomes completos do indivíduo como demonstrado abaixo:

> ... impetrado pelo advogado Paulo Jacob El Amm[PESSOA] em favor de ... apontando como autoridade coatora o Ministro Marco Aurélio Bellizze[PESSOA] do Superior Tribunal de Justiça ...

Classe Organização

Essa classe está relacionada com instituições privadas, públicas, sociais, organismos do governo e entre outros.

> ... os demandantes foram funcionários da Petrobrás[ORGANIZAÇÃO]. Partindo desta premissa ... devido o Instituto Nacional de Seguro Social (INSS)[ORGANIZAÇÃO] interpor ... agravante que o Superior Tribunal de Justiça[ORGANIZAÇÃO] negou provimento ...

Classe Local

A classe de localização foi estabelecida para abranger referências geográficas, como nomes de municípios, ruas, países ou objetos com referências geográficas. No entanto, não

foi utilizado sobreposição nas marcações, ou seja, caso um nome de lugar seja utilizado para se referir a um ente organizacional, sua classe será definida como organização e não como local.

> ... por volta das 04h45, na rua Antônio Dias Adorno, nº 264, Vila Nogueira[LOCAL], nesta cidade e comarca ... de distância de Brasília - DF[LOCAL], afirma, ainda ... que instituiu no Município de Conceição dos Ouros[ORGANIZAÇÃO], a Contribuição para Custeio ...

Classe Data

Para a entidade classe foi designada para capturar todas as referências temporais no texto, ou seja, não foi abrangido somente datas completas no formato dd/mm/yyyy, mas também referências parciais e textuais que permitem situar um evento temporal.

> ... decisão proferida em 13/10/2009[DATA], pelo Ministro ... processo em 7 de novembro de 2011[DATA], liberando-o para ser julgado ...

Classe Legislação

Para essa classe foi considerado normas em um sentido amplo, contando com portarias, leis federais, estaduais, municipais, estatutos, súmulas, e etc.

> ... disposto no art. 100, § 1º e § 4º, da Constituição federal[LEGISLACAO]. Transcrevo a ... nos termos das Súmulas 279 e 454/STF[LEGISLACAO].

Classe Documento

A classe entidade foi mapeada para considerar os eventos e marcos que estruturam o rito processual. Com isso tornou-se uma classe abrangente, possuindo em sua estrutura recursos, ações e remédios constitucionais, como habeas corpus, atos decisórios entre outras peças relevantes, como parecer, minutas etc.

> ... a concessão da ordem de habeas corpus[DOCUMENTO], para que se determine ... pedido ... O reclamante, em agravo regimental[DOCUMENTO], destaca ...

Classe Crime

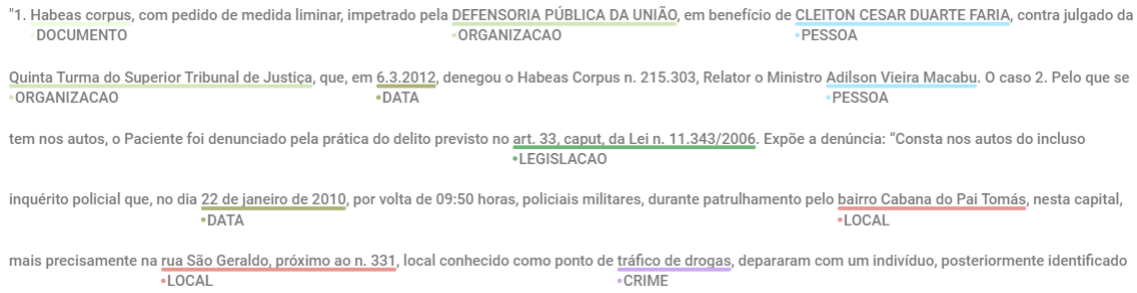
Por fim, a classe crime foi definida para extrair as infrações cometidas. Um exemplo pode ser visto logo abaixo:

> ... previsto no art. 297 do CP falsificação de documento público[CRIME]), à pena de 2 anos e ...pela suposta prática do crime de estelionato[CRIME] previsto no ...

3.3.2 Divisão dos Dados e tokenização

Concluída a etapa de anotação (Figura 6), os dados exportados passaram por transformações para ficar compatíveis com a biblioteca *datasets* do *Hugging Face*. Logo após, os dados foram divididos de forma estratificada em três conjuntos distintos para garantir uma avaliação robusta e imparcial do modelo, seguindo a proporção clássica da literatura: 70% treinamento, 15% validação e 15% testes.

Figura 6 – Exemplo de marcações de rótulos com doccano



Fonte: Autor

Em seguida, as classes semânticas foram convertidas para o esquema de rotulação IOB, na qual cada entidade foi delimitada por um rótulo de início, por exemplo, B-PESSOA, I-PESSOA, enquanto os tokens que não pertencem a nenhuma categoria recebem o rótulo "O". Essa transformação expandiu o conjunto de classes para um total de 15 rótulos distintos que serão utilizados na fase de *fine-tunning*.

Por fim, realizou-se a tokenização dos dados utilizando o método *WordPiece*, convertendo, assim, as sequências de palavras em uma representação numérica. Para esse processo, a opção de truncamento foi ativada para um comprimento máximo de 512 tokens e o preenchimento (*padding*) para garantir que todas as sequências de entrada tivessem um tamanho uniforme.

3.4 *Fine-Tuning*

O modelo base pré-treinado selecionado foi o *neuralmind/bert-base-portuguese-cased*, uma implementação da arquitetura *Transformer* extensivamente treinada para o português do Brasil. Para a realização dos experimentos, foi utilizado o ambiente de desenvolvimento Google Colaboratory, com a linguagem de programação *Python*. O treinamento foi conduzido com o auxílio da classe *Trainer*, fornecida pela biblioteca

Transformers, utilizando os argumentos listados na Tabela 4. No caso específico da tarefa de classificação de texto, foi adotado um *batch_size* igual a 16.

Tabela 4 – Parâmetros utilizados no treinamento do modelo na abordagem NER

Argumento	**Valor**
`num_train_epochs`	`4`
`per_device_train_batch_size`	`8`
`per_device_eval_batch_size`	`8`
`eval_strategy`	`"epoch"`
`save_strategy`	`"epoch"`
`learning_rate`	`2e-5`
`weight_decay`	`0.01`
`load_best_model_at_end`	`True`
`metric_for_best_model`	`"f1"`

3.5 Avaliação do Modelo

Para avaliação quantitativa deste trabalho, foram adotadas as métricas de Precisão, *Recall* e *F1-Score* para ambos os casos. Mas, o cálculo dessa métrica para o NER apresenta uma particularidade, pois um acerto não depende apenas da classificação correta da classe, mas também da delimitação exata dos seus limites de texto.

Para isso, foi utilizada a biblioteca *seqeval*, na qual é uma referência do script de avaliação oficial das competições CoNLL-2002 e CoNLL-2003. Logo, a *seqeval* analisa a sequência de etiqueta no formato IOB, respeitando rigorosamente os critérios definidos por Sang e Meulder (2003).

4 Resultados e Discussão

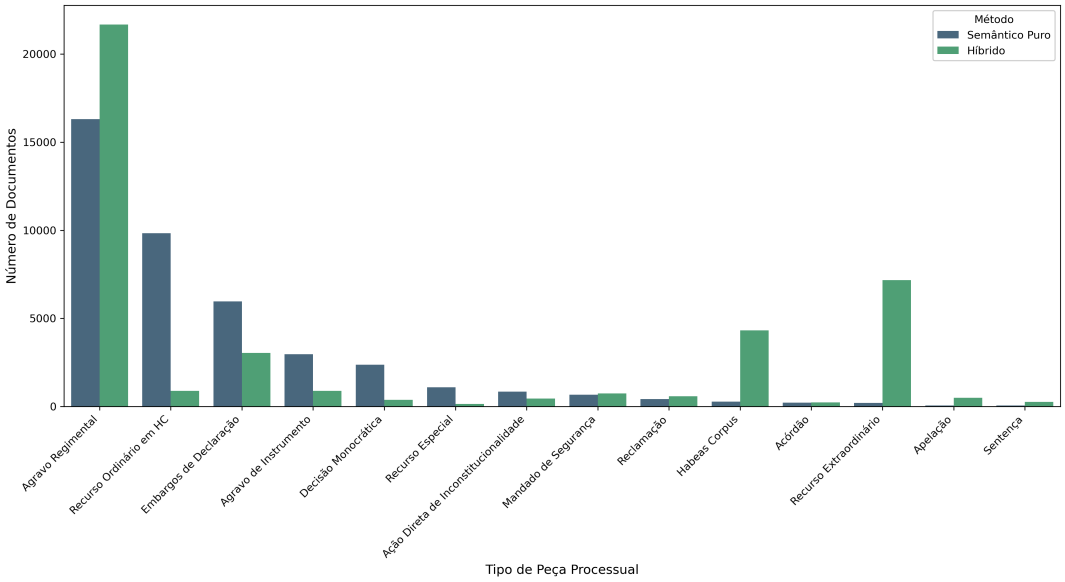
Esse capítulo apresenta os resultados obtidos nos experimentos descritos na metodologia, já acompanhados das respectivas discussões, sendo dividido em duas seções principais correspondente às tarefas centrais desse trabalho. A primeira seção apresenta os resultados da Classificação de Texto, enquanto a segunda detalha o desempenho do modelo de Reconhecimento de Entidade Nomeadas (NER). Para cada tarefa, serão apresentadas as métricas de precisão, *recall* e *F1-Score* além das análises qualitativas.

4.1 Resultado da Classificação de Texto

4.1.1 Análise Comparativa dos Métodos de Rotulagem

Nesta seção, será apresentada a análise do comportamento da utilização da supervisão fraca no modelo. O objetivo é então compreender como cada método interpretou e caracterizou os corpus jurídicos, e qual teve melhor desempenho aplicado em um modelo BERTimbau. A Figura 7 apresenta uma comparação entre as distribuições de classes em ambas as abordagens.

Figura 7 – Frequência de Classes encontrado pelo processo de Supervisão Fraca



Fonte: Autor

Analisando os resultados, é notória uma divergência entre as peças do tipo Agravo Regimental, Recursos Ordinários em HC, Recursos Extraordinário e Habeas Corpus entre ambos os métodos. Para a análise será utilizada a classe Recursos Ordinários em HC

conforme foi classificado pelo método semântico puro e avaliar, além de visualmente, o corpo textual e comparar com os resultados encontrados no modelo puro.

O método híbrido concordou com o método semântico em apenas 8% dos casos para essa classe. Já a maioria dos documentos foram classificados como Habeas Corpus (37,41%) simples. Adicionalmente, a segunda parcela mais significativo foi classificada como Agravo Regimental (21,57%) e Recurso Extraordinário (16,42%). Tal divergência pode estar relacionada a complexidade semântica do corpus e o modo como cada método funciona. Por exemplo, no documento com *ID* 15, foi classificado como Recursos Ordinários em HC pelo método semântico, mas como Habeas Corpus no método híbrido. O início do texto (apêndice A) explica essa ambiguidade.

> 'Trata-se de habeas corpus impetrado contra acórdão da Sexta Turma do Superior Tribunal de Justiça, nos autos do HC 348.763/SP, Rel. Ministro Rogério Schietti Cruz...'

Embora o documento se declarasse formalmente como um Habeas Corpus, a menção de certos termos, como "... impetrado contra acórdão da Sexta Turma ... ", cria-se uma sobreposição semântica com o protótipo Recursos Ordinário em HC (Sub-Seção 3.2.1). Assim, enquanto o método híbrido se ateve à declaração formal inicial, o semântico, que é dependente da qualidade e da distinção vetorial dos protótipos, foi influenciado pelo contexto da discussão recursal, ou seja, o peso das palavras se tornou mais próximo do protótipo Recursos Ordinário em HC do que Habeas Corpus.

Em contrapartida, as classes como Ação Direta de Inconstitucionalidade, Reclamação e Mandado de Segurança e Segurança apresentaram uma distribuição mais próxima entre ambos os métodos. Analisando a primeira classe citada, o método híbrido concordou em 47,81% dos casos proposto na análise semântica. Logo, isso indica que a linguagem e a estrutura desse tipo de peças são mais diretas e menos propensas a conter menções a múltiplos atos processuais que gerem confusão semântica.

4.1.2 Avaliação dos Modelos Classificadores

A avaliação dos modelos no conjunto de testes demonstra uma superioridade da abordagem Híbrida em relação à abordagem Semântica. Conforme mostrado na Tabela 5, o modelo treinado com os rótulos gerados pelo método Híbrido alcançou uma acurácia global de 98,31% e um *F1-Score* ponderado de 98,31%, apontando um ajuste de alta fidelidade do modelo durante a etapa de *fine-tuning*. Em comparação, a abordagem puramente Semântica resultou em um modelo com acurácia de 78,48% e *F1-Score* ponderado de 76,86%. Embora apresente um desempenho funcional interessante, a diferença notada de

performance entre as abordagens, evidencia o impacto crítico da metodologia de rotulação no resultado final.

Tabela 5 – Comparação de Desempenho por Classe entre Abordagem Híbrida e Abordagem Semântica

Classe	**Abordagem Híbrida**			**Abordagem Semântica**			**Suporte**
	Precisão	**Recall**	**F1-Score**	**Precisão**	**Recall**	**F1-Score**	
Acórdão	1,0000	0,8857	0,9394	0,0000	0,0000	0,0000	35
Agravo Regimental	0,9872	0,9957	0,9914	0,7937	0,8918	0,8399	3253
Agravo de Instrumento	0,9924	0,9850	0,9887	0,8454	0,5985	0,7009	133
Apelação	0,9730	0,9600	0,9664	0,0000	0,0000	0,0000	75
Ação Dir. Inconstitucionalidade	0,9828	0,8382	0,9048	0,8140	0,5426	0,6512	68
Decisão Monocrática	0,9623	0,8947	0,9273	0,6114	0,6233	0,6173	57
Embargos de Declaração	1,0000	0,9846	0,9923	0,7828	0,7400	0,7608	456
Habeas Corpus	0,9831	0,9815	0,9823	0,5385	0,1591	0,2456	650
Mandado de Segurança	0,9479	0,8198	0,8792	0,5077	0,2973	0,3750	111
Reclamação	1,0000	0,9659	0,9827	0,4000	0,2222	0,2857	88
Recurso Especial	0,5758	0,9048	0,7037	0,6475	0,5844	0,6143	21
Recurso Extraordinário	0,9944	0,9944	0,9944	0,5000	0,0278	0,0526	1077
Recurso Ordinário em HC	0,8819	0,9407	0,9104	0,8341	0,8879	0,8602	135
Sentença	0,8919	0,8250	0,8571	0,0000	0,0000	0,0000	40
Geral (Acurácia)	**0,9831**	**0,9831**	**0,9831**	**0,7848**	**0,7848**	**0,7848**	**6199**

Já uma análise detalhada do modelo Híbrido revela uma eficácia alta na maioria das classes. O modelo demonstrou uma capacidade de classificação altamente precisa para as categorias com maiores volumes de dados, como Recurso Extraordinário, que apresentou *F1-Score* de 0,9944%, Embargos de Declaração, que apresentou *F1-Score* de 0,9923%, e Agravo Regimental, que apresentou *F1-Score* de 0,9914. Tais resultados demonstram que a abordagem baseada em regras forneceu um aprendizado mais consistente e que foi eficientemente internalizado pelo modelo.

É válido destacar também o sucesso do modelo híbrido em diferenciar as nuances entre Habeas Corpus e Recursos Extraordinário em HC, os quais foram debatidas anteriormente. Apesar da sobreposição terminológica entre essas duas peças, o modelo foi capaz de aprender as características distintivas de cada uma, apresentando, para ambas, um *F1-Score* superior a 0,90. Já as classes que apresentaram o desempenho mais modesto foram Recurso Especial, com *F1-Score* igual a 0,7037, e Sentença, com *F1-Score* igual a 0,8537. Estes resultados abaixo dos demais podem estar correlacionados diretamente com o baixo número de amostras de suporte disponível para essas classes no conjunto de treinamento, que, consequentemente, pode ter limitado a capacidade do modelo de aprender a generalizar os seus padrões.

Por outro lado, o modelo treinado com rótulos semânticos, embora tivesse obtido seu melhor resultado na classe "Recursos Ordinários em HC"com um *F1-Score* de 0,8602, demonstrou várias dificuldades sistêmicas. A classe Habeas Corpus, por exemplo, foi severamente impactada pela metodologia de rotulagem, o que ficou evidente em sua precisão de 0,5385 e em seu *recall*, que foi extremamente baixo, de 0,1591. Esses resultados confirmam a hipótese anterior de que a definição do protótipo de "Recursos Ordinários

em HC"mostrou-se semanticamente dominante, atraindo e classificando incorretamente a maioria das instâncias de Habeas Corpus que possuíam terminologia recursal, resultando em uma substituição indevida da classe.

Portanto, a metodologia de rotulação Híbrida produziu um modelo que se ajustou de forma muito mais eficaz, alcançando uma performance global superior. No entanto, é válido ressaltar que, a abordagem Semântica, apesar de sua performance geral ter sido inferior a Híbrida, demonstrou um potencial notável para criação de rótulos por supervisão fraca, especialmente por ter mostrado que, em cenários em que a distinção conceitual/semântica seja o mais importante, ela pode ser a ferramenta adequada para a rotulagem.

4.2 Resultados do Reconhecimento de Entidades Nomeadas (NER)

O modelo NER alcançou um bom desempenho no conjunto de testes, atingindo um F1-Score geral de 0,8764, uma precisão de 0,8619 e um recall de 0,8913 (Tabela 6). É fundamental notar que esse *F1-Score* não é uma simples média dos resultados de cada classe. Em vez disso, ela corresponde a uma média ponderada que é calculada a partir da precisão e *recall* globais. Logo, as classes com mais frequência, como o Documento, a Legislação e a Organização, que juntas correspondem a aproximadamente 80% do *dataset*, influenciaram de forma predominante o resultado geral do modelo, explicando porque ele se manteve elevada mesmo com um desempenho inferior em duas classes.

Tabela 6 – Resultados Detalhados por Classe de Entidade no Conjunto de Teste

Classe	**Frequência (%)**	**Precisão**	**Recall**	**F1-Score**
Documento	41,0%	0,9152	0,9087	0,9119
Data	7,5%	0,8871	0,9167	0,9016
Organização	20,1%	0,8700	0,9223	0,8954
Pessoa	6,5%	0,9116	0,8590	0,8845
Legislação	20,4%	0,8383	0,8846	0,8608
Crime	3,9%	0,4729	0,6968	0,5634
Local	0,6%	0,5000	0,2069	0,2933
Geral	**100%**	**0,8619**	**0,8913**	**0,8764**

4.2.1 Análise de Performance por Classe

A análise detalhada da Tabela 6 permite avaliar de forma granular o comportamento do modelo para cada classe. As cinco primeiras classes apresentaram os melhores resultados do modelo, com destaque para a classe Documento e Data, que obtiveram *F1-Score* acima de 0,9. Em contrapartida, os piores resultados ficaram com Crime e Local, nos quais apresentaram *F1-Score* 0,5634 e 0,2933 respectivamente.

Essa disparidade nos resultados pode ser analisada em duas perspectivas. A primeira está relacionada com a frequência (Tabela 3) das classe no conjunto de dados de treinamento

do modelo. Assim, classes com poucos exemplos ou insignificantes podem não ter fornecido uma boa base de aprendizado para o modelo.

Já a segunda perspectiva é qualitativa e está relacionada com a estratégia de anotação. Conforme mencionado na metodologia (subseção 3.3.1), optou-se por não utilizar a sobreposição de marcação. A implicação direta disso é que o modelo foi forçado a priorizar uma classe em detrimento de outra com base no contexto funcional do termo. Por exemplo, em "Município de São Luís", o termo foi anotado como Organização por representar um ente jurídico, e isso evidencia que o modelo foi bem em associar o nome da cidade à sua função administrativa, e não a um lugar geográfico, mesmo que isso limite a sua capacidade de reconhecê-lo como Local na mesma frase.

Ainda na ótica qualitativa, foi realizada uma previsão do modelo em um texto inédito (Apêndice B), a fim de compreender seu comportamento em um cenário prático. O resultado da previsão foi filtrado para um score de confiança superior a 70%. A análise revela que as classes com padrões bem definidas obtiveram um desempenho notável, por exemplo, a classe Data (04 de Julho de 2025) foi extraída com um score de confiança 98,24%, e a classe Organização (Ministério Público Federal) com 98,24%.

Porém, a análise expõe também padrões de erros de delimitação de fronteira, por exemplo, o modelo identificou corretamente o "Artigo 37 da Constituição Federal que trata da" como uma classe LEGISLACAO com um score de 98,54%, mas ele incorretamente estendeu a anotação para incluir uma sentença explicativa subsequente.

Além disso foi observado alguns falsos positivos baseados em contextos. Por exemplo, o artigo "o" foi erroneamente classificado como pessoa, o que pode sugerir que o modelo tenha criado uma associação estatística que levou a uma predição incorreta devido a alta frequência com que precedem nomes próprios nos dados de treinamento. Por fim, muitos dos erros encontrados na predição poderiam ser mitigados em uma aplicação real através do ajuste do limiar de score para um valor mais elevado, como 90%, garantindo assim maior precisão na extração final.

5 Conclusão

Este trabalho apresentou o desenvolvimento e a validação de um sistema baseado em Processamento de Linguagem Natural voltado para o contexto jurídico brasileiro, explorando as tarefas de Reconhecimento de Entidades Nomeadas e Classificação de Texto sobre a base de dados *Iudicium Textum Dataset* utilizando um modelo *Transformer*. A metodologia empregada, que combinou abordagens heurísticas e semânticas, mostrou-se não apenas suficiente, mas fundamental para analisar e preparar os dados para a criação do modelo.

A aplicação da arquitetura Transformer, exemplificada pelo modelo BERTimbau, provou-se altamente adequada para o domínio jurídico em língua portuguesa. Na tarefa de classificação de peças, o modelo final da abordagem heurística alcançou uma acurácia de 98,31%, validando a eficácia da metodologia híbrida de rotulação que foi desenvolvida. Para a tarefa de NER, obteve-se um F1-Score geral de 87,6%, que demonstra a viabilidade da extração de informações a partir de textos complexos e reforçando o potencial da abordagem para a otimização da análise documental.

Outro ponto válido de ser destacado é o emprego da técnica de supervisão fraca para a criação dos gabaritos utilizados nesse estudo, que, diferentemente de muitos trabalhos da literatura no domínio jurídico brasileiro, partem de *datasets* previamente rotulados. Assim, esse estudo mostrou a viabilidade desse recurso em um cenário de recursos limitados, bem como analisou criticamente o impacto da qualidade e a natureza dos rótulos gerados no resultados finais, gerando *insights* importantes que podem nortear novos trabalhos na área.

5.1 Trabalhos futuros

Como trabalho futuro, destaca-se a possibilidade de aplicação de uma abordagem híbrida de supervisão fraca mais sensível ao contexto, utilizando regras de alta precisão, não só posicional, mas com uma segunda camada baseada em análise semântica. Essa segunda camada seria adicionada para analisar se o conteúdo do documento resolve as ambiguidades, como a distinção entre o ato processual principal do documento e os múltiplos recursos mencionados em seu corpo textual.

No que tange ao NER, a análise de performance revelou dificuldades do modelo em classes com baixo suporte de dados e alta sobreposição conceitual, como Local e Crime. Para mitigar esses desafios, uma direção para trabalhos futuros é a exploração da granularidade do esquema de anotação. Assim, propõe-se uma avaliação de um esquema com dois níveis de granularidade: uma grossa (*coarse-grained*), com categorias gerais, e um

fino (*fine-grained*), com subcategorias específicas. Esta abordagem segue a metodologia de Leitner, Rehm e Moreno-Schneider (2019), que desenvolveram um *dataset* jurídico com 7 classes gerais e 19 específicas, ajudando a padronização metodológica de pesquisa dos casos envolvendo domínio jurídico alemão.

Por fim, seria fundamental explorar o uso de anotações sobrepostas, permitindo que o um mesmo termo seja, por exemplo, classificado como Local e Organização. A hipótese aqui levantada é que, ao permitir essa flexibilidade, a performance em classe que naturalmente se sobrepõem aumentaria significativamente seus resultados, alinhando aos encontrados na literatura.

Referências

BA, J. L.; KIROS, J. R.; HINTON, G. E. Layer normalization. In: *Advances in Neural Information Processing Systems*. [S.l.: s.n.], 2016. v. 29, p. 262–270. Citado na página 17.

BATISTA, H. H.; NASCIMENTO, A. C.; MELO, R. F.; MIRANDA, P. B.; MALDONADO, I. W.; FILHO, J. L. C. A comparative analysis of text embedding approach to extract named entities in portuguese legal documents. In: SBC. *Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*. [S.l.], 2021. p. 221–232. Citado na página 15.

CHALKIDIS, I.; FERGADIOTIS, M.; MALAKASIOTIS, P.; ALETRAS, N.; ANDROUTSOPOULOS, I. Legal-Bert: The muppets straight out of law school. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, 2020. p. 2971–2981. Disponível em: <https://aclanthology.org/2020.findings-emnlp.268>. Citado na página 15.

Conselho Nacional de Justiça. *Justiça em números 2024*. [S.l.]: Conselho Nacional de Justiça, 2024. 448 p. ISBN 978-65-5972-140-5. Citado na página 13.

Conselho Nacional de Justiça. *Resolução nº 615, de 11 de março de 2025*. 2025. Disponível em: <https://rm.coe.int/resolucao-cnj-615-ia/1680b51b65>. Citado na página 14.

DARJI, H.; MITROVIĆ, J.; GRANITZER, M. German BERT model for legal named entity recognition. In: VÖLP, M.; MARES, M.; WIESNER, V. (Ed.). *Proceedings of the 16th International Conference on Computational Semantics (IWCS 2023)*. Nancy, France: Association for Computational Linguistics, 2023. p. 142–152. Disponível em: <https://aclanthology.org/2023.iwcs-1.12>. Citado na página 16.

DAS, S.; DAS, D. Natural language processing (nlp) techniques: Usability in human-computer interactions. In: *2024 6th International Conference on Natural Language Processing (ICNLP)*. [S.l.: s.n.], 2024. p. 783–787. Citado na página 13.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. [s.n.], 2019. p. 4171–4186. Disponível em: <https://aclanthology.org/N19-1423.pdf>. Citado 4 vezes nas páginas 20, 21, 22 e 28.

GRISHMAN, R.; SUNDHEIM, B. Message Understanding Conference- 6: A brief history. In: *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*. [s.n.], 1996. Disponível em: <https://aclanthology.org/C96-1079/>. Citado 2 vezes nas páginas 25 e 30.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778. Citado na página 17.

HENDRYCKS, D.; BURNS, C.; CHEN, A.; BALL, S. *CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review*. 2021. Disponível em: <https://arxiv.org/abs/2103.06268>. Citado na página 23.

KERAGHEL, I.; MORBIEU, S.; NADIF, M. Recent advances in named entity recognition: A comprehensive survey and comparative study. *Computation and Language*, 2024. Citado na página 25.

KERAGHEL, I.; MORBIEU, S.; NADIF, M. A survey on recent advances in named entity recognition. *arXiv preprint arXiv:2401.10825*, 2024. Citado na página 26.

LEITNER, E.; REHM, G.; MORENO-SCHNEIDER, J. Fine-grained named entity recognition in legal documents. In: SPRINGER. *International conference on semantic systems*. [S.l.], 2019. p. 272–287. Citado na página 41.

RAMSHAW, L. A.; MARCUS, M. P. Text chunking using transformation-based learning. In: *Third Workshop on Very Large Corpora*. [s.n.], 1995. Disponível em: <https://www.aclweb.org/anthology/W95-0107>. Citado na página 25.

RATNER, A.; BACH, S. H.; EHRENBERG, H.; FRIES, J.; WU, S.; RÉ, C. Snorkel: Rapid training data creation with weak supervision. In: *Proceedings of the VLDB endowment. International conference on very large data bases*. [S.l.: s.n.], 2017. v. 11, n. 3, p. 269. Citado na página 24.

REIMERS, N.; GUREVYCH, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019. Citado 2 vezes nas páginas 22 e 23.

SANG, E. F. T. K.; MEULDER, F. D. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In: *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*. Edmonton, Canada: Association for Computational Linguistics, 2003. p. 142–142. Disponível em: <https://aclanthology.org/W03-0407>. Citado na página 34.

SILVEIRA, R.; PONTE, C.; ALMEIDA, V.; PINHEIRO, V.; FURTADO, V. Legalbert-pt: A pretrained language model for the brazilian portuguese legal domain. In: NALDI, M. C.; BIANCHI, R. A. C. (Ed.). *Intelligent Systems*. Cham: Springer Nature Switzerland, 2023. p. 268–282. ISBN 978-3-031-45392-2. Citado na página 16.

SOUSA, A. W.; FABRO, M. D. D. Iudicium textum dataset uma base de textos jurıdicos para nlp. In: *XXXIV Simpósio Brasileiro de Banco de Dados: Dataset Showcase Workshop, SBBD*. [S.l.: s.n.], 2019. p. 1–11. Citado na página 27.

SUN, H.; XU, C.; SUOMINEN, H. Analyzing the granularity and cost of annotation in clinical sequence labeling. In: *Proceedings of the 2nd Workshop on Natural Language Processing for Medical Conversations (NLPMC)*. Online: Association for Computational Linguistics, 2021. p. 12–22. Disponível em: <https://aclanthology.org/2021.nlpmc-1.2>. Citado na página 23.

Thomson Reuters Institute. *2022 Report on the State of the Legal Market: A challenging road to recovery*. 2022. Accessed: 2025-07-09. Disponível em: <https://www.thomsonreuters.com/en-us/posts/wp-content/uploads/sites/20/2022/01/State-of-Legal-Market-Report_Final.pdf>. Citado na página 13.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. *Advances in neural information processing systems*, v. 30, 2017. Citado 4 vezes nas páginas 14, 17, 18 e 19.

ZHONG, J.; XUAN, Z.; WANG, K.; CHENG, Z. A bert-span model for named entity recognition in rehabilitation medicine. *BMC Medical Informatics and Decision Making*, BioMed Central, v. 23, n. 1, p. 1–13, 2023. Citado na página 15.

Apêndices

APÊNDICE A – Exemplo de Documento para Análise Qualitativa

Este exemplo representa um documento jurídico submetido à análise semântica por dois métodos distintos.

ID do Documento: 15
Rótulo no Modelo Híbrido: *Habeas Corpus*
Previsão Semântica Pura: *Recurso Ordinário em HC*

Texto Completo:

> *'Trata-se de habeas corpus impetrado contra acórdão da Sexta Turma do Superior Tribunal de Justiça, nos autos do HC 348.763/SP, Rel. Ministro Rogério Schietti Cruz. Consta dos autos, em síntese, que (a) o paciente foi preso preventivamente pela suposta prática do crime de extorsão (art. 158, § 1º, do Código Penal); (b) inconformada, a defesa impetrou habeas corpus no Tribunal de Justiça do Estado de São Paulo, que denegou a ordem, e, na sequência, outro HC no Superior Tribunal de Justiça; (c) enquanto se aguardava o julgamento da impetração no STJ, o juízo de primeiro grau, ao proferir sentença condenatória, manteve a segregação cautelar, razão pela qual o Ministro Relator julgou prejudicado o referido habeas corpus; (d) contra essa decisão, foi interposto agravo regimental, que foi desprovido, em acórdão assim ementado: [...] 1. Prolatada sentença condenatória, por meio da qual o Juízo singular empreendeu nova avaliação sobre os fundamentos suscitados para a imposição da segregação cautelar (art. 387, § 1º, do CPP), tais razões devem ser submetidas ao crivo do Tribunal a quo. 2. Agravo regimental não provido. Neste habeas corpus, os impetrantes alegam, em suma, que (a) a superveniência da sentença condenatória não importa em perda de objeto do habeas corpus no STJ, uma vez que houve a mera reiteração dos fundamentos utilizados para a decretação da cautelar; (b) não estão presentes os pressupostos autorizadores da prisão preventiva, descritos no art. 312 do Código de Processo Penal; (c) o caso comporta a imposição de medidas alternativas à prisão, previstas no art. 319 do Código de Processo Penal. Requerem, assim, a expedição de alvará de soltura para que o paciente possa recorrer em liberdade, com ou sem aplicação de outras medidas cautelares. O pedido de liminar foi indeferido. Em parecer, o Ministério Público*

Federal manifesta-se pelo não conhecimento da impetração e, no mérito, pela denegação da ordem. Há pedido de intimação para fins de sustentação oral. É o relatório.'

APÊNDICE B – Estudo de Caso: Análise de Previsão do Modelo

A seguir, apresenta-se um trecho textual utilizado como entrada no modelo para fins de análise de reconhecimento de entidades nomeadas (NER):

> Conforme a decisão proferida em 04 de julho de 2025, o Juiz Federal Carlos Almeida, da Vara Cível de São Luís, determinou o prosseguimento da Ação Civil Pública. O processo foi iniciado pelo Ministério Público Federal (MPF) com base no Artigo 37 da Constituição Federal, que trata da impessoalidade. O réu, a empresa TechCorp S.A., deverá apresentar sua apelação no prazo legal.

Previsão do modelo com confiança maior ou igual a 70%

> Conforme a decisão proferida em **[04 de julho de 2025,]**(DATA) **[o]**(PESS) Juiz Federal **[Carlos Almeida,]**(PESS) da **[Vara Cível de São Luís, determinou]**(ORG) o **prosseguimento da**[DOC] **[Ação Civil Pública. O]**(DOC) processo foi iniciado pelo **[Ministério Público Federal]**(ORG) **[(MPF)]**(ORG) com base no **[Artigo 37 da Constituição Federal, que trata da]**(LEG) **[impessoalidade]**(CRM). O **[réu,]**(PESS) **[a]**(ORG) **[empresa]**(ORG) **[TechCorp S.A., deverá]**(ORG) apresentar sua **[apelação no]**(DOC) prazo legal.

Legenda das siglas:

- **PESS** — Pessoa
- **ORG** — Organização
- **DOC** — Documento
- **LEG** — Legislação
- **CRM** — Crime
- **DATA** — Data