UNIVERSIDADE ESTADUAL DO MARANHÃO

Graduação em Engenharia de Computação

Jonas Carvalho de Sousa Neto

Similaridade Semântica entre Processos Judiciais: Uma Abordagem para Identificação de Litigância Predatória

São Luís - MA

2025

Jonas Carvalho de Sousa Neto

Similaridade Semântica entre Processos Judiciais: Uma Abordagem para Identificação de Litigância Predatória

Trabalho de conclusão apresentado como requisito parcial para obtenção do título de Engenheiro de Computação.

Curso de Engenharia de Computação

Universidade Estadual do Maranhão - UEMA

Orientador: Adrielson Ferreira Justino

São Luís - MA

2025

Sousa Neto, Jonas Carvalho de
Similaridade Semântica entre Processos Judiciais: Uma Abordagem para Identificação de Litigância Predatória. / Jonas Carvalho de Sousa Neto. – São Luis, MA, 2025.
64 f

TCC (Graduação em Engenharia de Computação) - Universidade Estadual do Maranhão, 2025.

Orientador: Adrielson Ferreira Justino.

1.Similaridade Semântica. 2.Representações Vetoriais. 3.Litigância Predatória. 4.Recuperação de Informação. I. Título.

CDU: 004.8:34

Elaborado por Cássia Diniz - CRB 13/910

Jonas Carvalho de Sousa Neto

Similaridade Semântica entre Processos Judiciais: Uma Abordagem para Identificação de Litigância Predatória

Trabalho de conclusão apresentado como requisito parcial para obtenção do título de Engenheiro de Computação.

São Luís - MA, 17 de fevereiro de 2025:

Documento assinado digitalmente
gov.br
ADRIELSON FERREIRA JUSTINO
Data: 21/02/2025 18:45:45-0300
Verifique em https://validar.iti.gov.br

Adrielson Ferreira Justino
Orientador - Universidade Federal do Oeste do Pará - UFOPA

Documento assinado digitalmente
gov.br
ANTONIO FERNANDO LAVAREDA JACOB JUNIOI
Data: 22/02/2025 09:12:06-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Antônio Fernando Lavareda Jacob Júnior
Examinador Interno - Universidade Estadual do Maranhão - UEMA

Documento assinado digitalmente
gov.br
FABRICIO ALMEIDA DO CARMO
Data: 21/02/2025 19:22:28-0300
Verifique em https://validar.iti.gov.br

Me. Fabrício Almeida do Carmo
Examinador Externo - Universidade Federal do Maranhão - UFMA

Agradecimentos

Dedico este trabalho a Deus, por me conceder a força e a motivação necessárias para trilhar o caminho dos estudos. Ao meu pai, Gilberto Carlos, à minha madrasta, Charliene Gomes, e à minha irmã, Safira Gomes, por todo o incentivo e apoio incondicional; sem cada um de vocês, este trabalho não teria sido possível. Aos meus amigos da graduação, que estiveram ao meu lado, me apoiaram e compartilharam desafios que, sozinho, eu não teria superado. Aos membros do LINCProg, em especial ao professor Dr. Antônio Jacob e ao meu orientador Adrielson Justino, por todo o suporte e direcionamento, que foram essenciais para minha evolução acadêmica.

"Se algum dia se sentir desmotivado ou achar que não é bom o suficiente, incendeie seu coração, enxugue as lágrimas e siga em frente." Kyojuro Rengoku (*Kimetsu no Yaiba: Mugen Train*)

Resumo

Segundo o Conselho Nacional de Justiça, o Poder Judiciário Brasileiro recebe anualmente mais de 30 milhões de processos judiciais. Esse volume excessivo de dados evidencia a necessidade de soluções automatizadas para aumentar a eficiência e a produtividade do sistema judiciário. Nesse cenário, um dos desafios enfrentados é a litigância predatória, caracterizada pela protocolação de ações judiciais de má-fé, nas quais as mesmas partes (autor, réu e advogado) submetem múltiplas demandas da mesma problemática, gerando esgotamento dos recursos do judiciário. Em 2020, cerca de 30% dos processos de Direito Civil e do Consumidor foram classificados como litigância predatória, gerando um custo mínimo de R$ 10,7 bilhões ao Judiciário. Além dos prejuízos financeiros, essa prática sobrecarrega magistrados e servidores, aumentando o tempo de tramitação de processos legítimos e comprometendo a eficiência do sistema. Este trabalho investiga a utilização de *embeddings* baseados em *Transformers*, como o *Bidirectional Encoder Representations from Transformers*, para análise de padrões de litigância predatória por meio de similaridade semântica. Esses modelos são capazes de capturar o significado semântico de palavras e frases, considerando o contexto em que estão inseridas. Para a construção e avaliação da pesquisa, foram utilizadas técnicas de Recuperação de Informação, que permite medir a eficácia dos modelos na identificação de casos semanticamente semelhantes. Foram avaliados quatro modelos de linguagem (*BERTikal*, *BERTimbau*, *BumbaBERT Small* e o *RoBERTa ptBR*), dentre os quais o *BERTimbau* obteve o melhor desempenho (*Recall@k* de 69,23% e *MAP* de 35,58%), seguido pelo *BumbaBERT Small* (*Recall@k* de 61,54% e *MAP* de 31,13%). Outros modelos testados apresentaram desempenho inferior. Estes resultados demonstram a viabilidade da proposta, contribuindo para a automação de tarefas no Judiciário e avanços nas técnicas de Processamento de Linguagem Natural.

Palavras-chave: Similaridade semântica, Representações vetoriais, Litigância predatória, Recuperação da informação.

Abstract

According to the National Council of Justice, the Brazilian Judiciary receives more than 30 million court cases every year. This excessive volume of data highlights the need for automated solutions to increase the efficiency and productivity of the judicial system. In this scenario, one of the challenges faced is predatory litigation, characterized by the filing of lawsuits in bad faith, in which the same parties (plaintiff, defendant and lawyer) submit multiple claims on the same issue, generating exhaustion of judicial resources. In 2020, around 30% of civil and consumer law cases were classified as predatory litigation, generating a minimum cost of R$ 10.7 billion to the judiciary. In addition to the financial losses, this practice overloads judges and civil servants, increasing the time it takes to process legitimate cases and compromising the efficiency of the system. This paper investigates the use of *embeddings* based on *Transformers*, such as *Bidirectional Encoder Representations from Transformers*, to analyze predatory litigation patterns by means of semantic similarity. These models are capable of capturing the semantic meaning of words and phrases, considering the context in which they are inserted. Information Retrieval techniques were used to construct and evaluate the research, which allows the effectiveness of the models to be measured in identifying semantically similar cases. Four language models were evaluated (*BERTikal*, *BERTimbau*, *BumbaBERT Small* and *RoBERTa ptBR*), of which *BERTimbau* had the best performance (*Recall@k* of 69.23% and *MAP* of 35.58%), followed by *BumbaBERT Small* (*Recall@k* of 61.54% and *MAP* of 31.13%). Other models tested performed less well. These results demonstrate the viability of the proposal, contributing to the automation of tasks in the Judiciary and advances in Natural Language Processing techniques.

Keywords: Semantic similarity, Vector representations, Predatory litigation, Information retrieval.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AP	*Average Precision*
BERT	*Bidirectional Encoder Representation from Transformers*
CRISP-DM	*CRoss Industry Standard Processing for Data Mining*
CSV	*Comma-Separated Values*
GPT	*Generative Pre-Trained Transformer*
GCP	*Google Cloud Platform*
MAP	*Mean Average Precision*
ML	*Machine Learning*
MRR	*Mean Reciprocal Rank*
NDCG	*Normalized Discounted Cumulative Gain*
OAB	*Ordem dos Advogados do Brasil*
PLN	*Processamento de Linguagem Natural*
RoBERTa	*Robustly Optimized BERT Approach*
SCI	*Suprema Corte da Índia*
SI	*Sistemas Inteligentes*
SIGIR	*Special Interest Group on Information Retrieval*
TF-IDF	*Term Frequency - Inverse Document Frequency*
TJMA	*Tribunal de Justiça do Maranhão*
TREC	*Text Retrieval Conference*
UEMA	*Universidade Estadual do Maranhão*

Sumário

1 INTRODUÇÃO

O desenvolvimento tecnológico, impulsionado pela Indústria 4.0, tem transformado diversos setores da sociedade ao promover a interação entre máquinas, sistemas e ativos por meio de redes de computadores inteligentes (BARRETO, 2019). No âmbito do Poder Público, essa transformação também se fez presente, especialmente no Judiciário — um sistema complexo e moroso que lida com os interesses conflitantes das partes envolvidas em processos judiciais, como réus, autores e advogados (KABIR; ALAM, 2023). A digitalização dos processos, foi impulsionada pela implementação do Processo Judicial Eletrônico (PJe), um sistema responsável por armazenar e organizar os documentos processuais (FEDERAL, 2011). No entanto, o grande volume de dados gerados pelo Judiciário brasileiro ainda representa um desafio. De acordo com o relatório Justiça em Números do Conselho Nacional de Justiça (CNJ)[1], em 2023 foram iniciados 35 milhões de novos processos, somando-se aos já existentes, totalizando 83,8 milhões de processos em tramitação. Nesse contexto, embora a digitalização tenha modernizado os procedimentos, ainda existem limitações no uso de tecnologias da informação capazes de lidar de forma eficaz com o grande volume de documentos judiciais (FILHO; JUNQUILHO, 2018).

Além disso, o Judiciário brasileiro enfrenta desafios administrativos que podem comprometer a segurança jurídica (ASPERTI et al., 2019). Entre eles, destaca-se o fenômeno da litigância predatória, também conhecida como demanda predatória, que caracteriza o uso abusivo do direito de ação com o objetivo de obter vantagens indevidas (MAGALHÃES; SOUSA et al., 2024). Essa prática não apenas afeta a segurança jurídica, mas também sobrecarrega o sistema, aumentando os custos e o tempo de tramitação de processos legítimos. Em 2020, cerca de 30% dos processos de Direito Civil e do Consumidor foram classificados como litigância predatória, gerando um custo mínimo de R$ 10,7 bilhões ao Judiciário (JOSÉ; CLEMENTINO, 2024). Além disso, a litigância predatória afeta diretamente o acesso à justiça, garantido pelo artigo 5º, inciso XXXV, da Constituição Federal de 1988, que assegura a todos o direito de buscar a proteção judicial em caso de ameaça ou lesão a direitos (Jusbrasil)[2]. No entanto, muitos desses processos são instaurados de forma irregular, muitas vezes por pessoas juridicamente vulneráveis que desconhecem a natureza predatória de suas ações (JOSÉ; CLEMENTINO, 2024).

Um exemplo da prática de litigância predatória foi identificado no Tribunal de Justiça do Mato Grosso do Sul, onde mais de 27 mil ações foram protocoladas por um único advogado, muitas delas com alegações genéricas e sem provas anexadas (JOSÉ;

[1] <https://www.cnj.jus.br/wp-content/uploads/2024/05/justica-em-numeros-2024-v-28-05-2024.pdf>

[2] <https://www.jusbrasil.com.br/topicos/10729607/inciso-xxxv-do-artigo-5-da-constituicao-federal-de-1988>

CLEMENTINO, 2024). Além disso, uma nota técnica do Centro de Inteligência do Tribunal de Justiça de Minas Gerais destacou que boa parte dos processos julgados mensalmente apresenta elementos de litigância artificialmente criada, com custos que podem alcançar R$ 25 bilhões anuais (MAGALHÃES; SOUSA et al., 2024). Apesar dos esforços do Judiciário para combater essa prática, como a criação do Incidente de Resolução de Demandas Repetitivas (IRDR) e a utilização de painéis de monitoramento como o dos Grandes Litigantes do CNJ[3], a identificação de casos de litigância predatória ainda é complexa, exigindo uma análise criteriosa do contexto e do conteúdo dos processos (SILVA; ZUCOLOTO; BARBOSA, 2013).

Outro ponto de destaque é o impacto do período pandêmico, que impôs novos desafios ao Poder Público, exigindo uma mudança na abordagem no tratamento dos processos (ARAÚJO; GABRIEL; PORTO, 2022). Nesse cenário, surgiu o Programa Justiça 4.0, uma iniciativa que promove o uso de Tecnologias da Informação e Comunicação (TICs) nos Tribunais de Justiça, garantindo eficiência e facilidade nas operações do Judiciário (RAMPIM; IGREJA, 2022). O programa busca alinhar-se ao contexto de desenvolvimento tecnológico atual, estabelecendo parcerias entre tribunais, universidades e empresas especializadas em tecnologia para desenvolver ferramentas baseadas em Inteligência Artificial (IA) e automação (BRAGANÇA; BRAGANÇA, 2019). Nesse contexto, uma das medidas adotadas para atender à crescente demanda por inovação no setor jurídico brasileiro tem sido o estabelecimento de parcerias estratégicas de vários tribunais com universidades e empresas especializadas em tecnologia para desenvolver ferramentas baseadas em IA[4].

Um exemplo concreto dessa colaboração é o Acordo de Cooperação Técnica n.º 002/2021, firmado entre a Universidade Estadual do Maranhão (UEMA) e o Tribunal de Justiça do Maranhão (TJMA). Essa parceria técnico-científica tem como foco o desenvolvimento de soluções baseadas em IA e automação, visando promover eficiência operacional nos procedimentos relacionados à aplicação da lei (CARMO, 2024). No contexto do combate à litigância predatória, o TJMA desenvolveu o robô Nirie, uma ferramenta automatizada que realiza buscas estruturadas e por expressões regulares em processos judiciais para identificar possíveis casos de demandas predatórias. No entanto, apesar de sua utilidade, o robô apresenta limitações, como a dependência de busca estruturada e a incapacidade de analisar o contexto semântico dos processos.

Neste sentido, este trabalho apresenta uma abordagem para identificar padrões de litigância predatória por meio de similaridade semântica textual de processos jurídicos, utilizando mecanismos de Processamento de Linguagem Natural (PLN). Essas técnicas, amplamente empregadas no desenvolvimento de TICs, têm o potencial de contribuir para a análise contextual e de conteúdo dos processos, auxiliando na detecção de práticas abusivas

[3] <https://justica-em-numeros.cnj.jus.br/painel-litigantes/>
[4] <https://www.cnj.jus.br/tribunal-e-universidade-debatem-parceria-de-tecnologia-no-ms/>

e na redução da sobrecarga do Judiciário (CNJ, 2025). Essa iniciativa está alinhada aos princípios do Programa Justiça 4.0, visando realizar a aplicação de tecnologias inovadoras para garantir maior eficiência e celeridade na prestação jurisdicional.

Para isso foi investigado a viabilidade e implementação de um *framework* baseado em Recuperação de Informação (RI) para apoiar a análise de demandas predatórias no sistema judiciário. Este estudo, poderá ser utilizado como um complemento ao robô Nirie do TJMA, visando melhorias nas predições referentes a litigiosidade predatória nos processos judiciais. Além disso, este estudo busca contribuir para pesquisas de PLN aplicadas ao Judiciário, através da exploração de soluções inovadoras aos desafios específicos do setor.

1.1 Objetivo Geral

Investigar o uso de *embeddings* contextuais na recuperação de informação, com base na similaridade semântica, para otimizar a busca de documentos relevantes e apoiar a análise de litigância predatória.

1.2 Objetivos Específicos

1. Identificar e comparar modelos de linguagem na literatura, considerando sua aplicabilidade ao contexto jurídico;

2. Gerar representações vetoriais (*embeddings*) a partir dos modelos de linguagem selecionados;

3. Desenvolver e implementar um *framework* experimental para avaliar a eficácia das representações vetoriais na tarefa de recuperação de informação através da busca por similaridade semântica em processos.

4. Sintetizar os resultados em produções técnico-científicas, como relatórios técnicos, e para divulgação em conferências ou periódicos.

1.3 Organização do Trabalho

O presente trabalho foi distribuído em 6 capítulos. No Capítulo 1, é apresentado o contexto no qual o estudo está inserido e a apresentação do objeto de pesquisa que são as demandas predatórias no Judiciário. O Capítulo 2 dispõe do referencial teórico relacionado a litigiosidade e as técnicas escolhidas para compor o trabalho. No Capítulo 3, são dispostos trabalhos relacionados a representação vetorial, recuperação de informação e busca por similaridade textual que podem contribuir para esta pesquisa através da discussão das técnicas utilizadas e resultados obtidos. No Capítulo 4, são expostos os materiais e métodos

utilizados na construção do trabalho, dando ênfase a metodologia *Cross Industry Standard Processing for Data Mining* (CRISP-DM). No Capítulo 5, é apresentado a discussão sobre os resultados obtidos a partir do *framework* desenvolvido. Por fim, no Capítulo 6 são apresentadas as considerações finais, expondo os impactos, as dificuldades e as expectativas resultantes deste trabalho, além de propor melhorias e pontos que podem ser estudados em trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Litigância Predatória

A litigiosidade, também conhecida por litigância, é um fenômeno social que remonta à Revolução Industrial e se intensificou com os modelos de produção atuais, baseados em uma sociedade de consumo em massa e inovações tecnológicas (MAGALHÃES; SOUSA et al., 2024). No âmbito jurídico, a litigância refere-se ao ato de litigar, ou seja, levar disputas ao Poder Judiciário. No entanto, quando esse direito é exercido de forma abusiva ou desonesta, surge a litigância predatória, caracterizada pela propositura de ações judiciais de má-fé, nas quais as mesmas partes (autor, réu e advogado) repetem demandas sem fundamento real, visando obter vantagens indevidas ou sobrecarregar o sistema judiciário (JOSÉ; CLEMENTINO, 2024). Essa prática compromete a eficiência e a credibilidade do sistema de justiça, além de gerar impactos econômicos e sociais significativos.

A litigância predatória engloba duas ideias principais: a litigiosidade, que consiste em levar conflitos a diferentes instâncias do Poder Judiciário por meio de recursos judiciais; e a predação, que se refere ao consumo excessivo de recursos do Judiciário ou da parte contrária, impactando sua capacidade de defesa e a viabilidade de autuação (JOSÉ; CLEMENTINO, 2024). Embora algumas condutas demandem uma alocação considerável de recursos, nem sempre devem ser consideradas predatórias. A prática só é caracterizada como tal quando há uma intenção subversiva, manifestada por meio de demandas repetitivas, complexas ou fraudulentas (MAGALHÃES; SOUSA et al., 2024).

Essa prática gera impactos significativos no sistema judiciário, tanto em termos econômicos quanto operacionais. Dados coletados pelos Centros de Inteligência e pelos Núcleos de Monitoramento do Perfil de Demandas (NUMOPEDEs) indicam que, em média, 30% dos processos relacionados a esses assuntos consistem em litigância predatória artificialmente criada, ou seja, sem base em fatos. Considerando o custo médio de um processo judicial, estimado em R$ 8.270,12 (valor corrigido em 2022), o prejuízo mínimo causado ao erário em 2020 foi de R$ 10,7 bilhões na Justiça Estadual e R$ 2,1 bilhões nos Juizados Especiais. Esse curso é majoritariamente absorvido pelo Estado, já que quase 100% dessas ações são movidas sob o regime de justiça gratuita.

O relatório Justiça em Números de 2021[1] expõe que além dos prejuízos financeiros, a litigância predatória sobrecarrega magistrados e servidores, que dedicam tempo valioso à apreciação de demandas fraudulentas. Esse tempo poderia ser direcionado à resolução de litígios legítimos, reduzindo o tempo médio de tramitação dos processos e aumentando

[1] https://www.cnj.jus.br/wp-content/uploads/2021/09/relatorio-justica-em-numeros2021-12.pdf

a eficiência e a credibilidade do sistema de justiça. Consoante a isso, segundo o Centro de Inteligência da Justiça de Minas Gerais (CIJMG) a prática também dificulta a elaboração de estatísticas confiáveis, uma vez que muitos processos são cadastrados de forma indevida, seja por erro ou para ocultar abusos.

Uma das práticas mais comuns na litigância predatória é a manipulação de documentos textuais (MAGALHÃES; SOUSA et al., 2024). Os litigantes predatórios frequentemente redigem petições e recursos alterando pequenas partes do texto, como datas, valores ou detalhes específicos, para criar a ilusão de demandas distintas. Essa estratégia dificulta a identificação de processos aparentemente novos, mas que, na realidade, tratam de questões já julgadas ou sem fundamento legal (JOSÉ; CLEMENTINO, 2024). A análise desses documentos textuais, portanto, é fundamental para identificar práticas de má-fé e combater a litigância predatória.

O crescimento da informatização do processo judicial trouxe consigo um aumento expressivo no volume de ações judiciais, com padrões repetitivos e características padronizadas (MAGALHÃES; SOUSA et al., 2024). Diante disso, a utilização de técnicas de PLN podem ser úteis para detectar padrões textuais repetitivos em petições judiciais (RODRÍGUEZ; BEZERRA, 2020). Por conseguinte, a utilização de técnicas como a busca por similaridade semântica é capaz de retornar grupos de processos com alto grau de semelhança (COSTA, 2024). Dessa forma, a utilização de automações que combinam essas técnicas permite que os tribunais atuem de forma mais eficiente no combate a litígios predatórios (JOSÉ; CLEMENTINO, 2024).

2.2 Robô Nirie

O robô Nirie, desenvolvido pelo TJMA, é uma ferramenta destinada a combater a litigância predatória, realizando uma triagem sobre novos processos protocolados para identificar possíveis demandas predatórias. Conforme o relatório de funcionamento do robô, disponibilizado pelo Departamento de TI do TJMA, a análise é iniciada a partir da entrada de um novo processo.

O robô verifica, inicialmente, se há advogados entre as partes do processo. Caso não haja, o processo é marcado como “Sem pendências”, e a análise é encerrada. Caso contrário, o robô segue com a validação da petição inicial e a busca por prevenção, aplicando etiquetas específicas conforme os resultados obtidos, conforme detalhado na Tabela 1.

ID	Etiqueta	Descrição
0	SEM ETIQUETA	Quando não há pendências no processo (não há advogados entre as partes).
1	[TJMA] NIRIE DOC-PETICAO-INVALIDA	Aplicada quando a petição inicial é considerada inválida pelos modelos de classificação do CNJ e UEMA.
2	[TJMA] NIRIE PREV-PREVENTO-[número do processo ao qual é prevento]	Aplicada aos processos subsequentes que possuem características semelhantes a um processo prevento já identificado. O número do processo prevento é incluído na etiqueta.
3	[TJMA] NIRIE PREV-POSSIVEL-PREVENCAO	Aplicada ao primeiro processo prevento identificado, indicando uma possível prevenção.
4	[TJMA] NIRIE PREV-POSSIVEL-DEMANDA-PREDATORIA	Aplicada quando há indícios de demanda predatória, como petições inválidas ou idênticas em processos protocolados em um curto intervalo de tempo.
5	[TJMA] NIRIE PREV-PROCESSO-REFERENCIA	Aplicada quando o processo é identificado como referência para outros processos preventos.
6	[TJMA] NIRIE NAO-FOI-POSSIVEL-ANALISAR-DOCUMENTO	Aplicada quando ocorre um erro durante a validação da petição inicial, impedindo a conclusão da análise.
7	[TJMA] NIRIE NAO-FOI-POSSIVEL-ANALISAR-PREVENCAO	Aplicada quando ocorre um erro durante a busca de prevenção, impedindo a conclusão da análise.

Tabela 1 – Rótulos aplicados na base de dados pelo robô Nirie.
Fonte: Adaptado do relatório fornecido pelo Departamento de TI do TJMA (2024).

Após a verificação da presença de advogados, o robô realiza a validação da petição inicial. Para isso, o conteúdo da petição é submetido a dois modelos de classificação: um desenvolvido pelo CNJ e outro pela UEMA. A petição é considerada válida se pelo menos um dos modelos retornar um resultado positivo. Caso contrário, a etiqueta 1 é aplicada. Se ocorrer algum erro durante a validação, a etiqueta 6 é atribuída.

Em seguida, o robô realiza uma busca estruturada na base de dados do PJE, utilizando filtros predefinidos, como partes (polos ativo e passivo), assuntos, jurisdição, classe judicial, órgão julgador e advogados envolvidos. Essa busca é realizada em processos protocolados nos últimos 50 dias visando identificar processos semelhantes protocolados em um curto intervalo de tempo, indicando possíveis repetições ou abusos.

Se for encontrada correspondência, o robô aplica a Etiqueta 3 ao primeiro processo prevento protocolado, indicando que este é o processo de referência. Em seguida, a Etiqueta 2 é aplicada aos processos subsequentes que possuem características semelhantes, incluindo o número do primeiro processo prevento na etiqueta. No entanto, se houver indícios de

demanda predatória, a Etiqueta 4 é aplicada. Essa identificação é realizada por meio da busca de expressões regulares, focada na detecção de números de contratos de empréstimos consignados, permitindo avaliar se as petições iniciais representam casos semelhantes ou idênticos. Em um cenário excepcional, no qual a busca de prevenção não seja possível e impeça a conclusão da análise, a Etiqueta 7 é atribuída.

Mediante a estratégia de aplicação de rótulos, existem quatro casos específicos que exigem tratamentos particulares. A seguir, são analisados os seguintes cenários:

1. **Empréstimos consignados:** Este tipo de processo só pode ser analisado no 1º Grau do PJe. Nesse cenário, após a identificação da prevenção, o robô busca os números de contrato no conteúdo da petição inicial, utilizando técnicas de PLN, incluindo o uso de expressões regulares para realizar a tarefa. Caso o número do contrato seja o mesmo, é aplicada a Etiqueta 4.

2. **Agravo de instrumento:** Isso ocorre quando as decisões tomadas pelo juiz não põem fim à demanda. Esse instrumento processual é essencial para garantir a revisão de decisões que podem impactar o andamento ou o resultado do processo. Para este caso, só é realizada a análise caso o agravo esteja no 2º Grau do PJe. Nessas situações, o robô não realiza a busca por prevenção ao identificar que se trata de um agravo de instrumento.

3. **Processos com referência:** Também restritos à análise no 2º Grau do PJe, esses processos seguem a análise padrão do robô. Neste cenário, as referências desses processos são outras demandas com características semelhantes que servem como parâmetro para análise e julgamento. No entanto, caso os resultados da busca inicial sejam insuficientes, o sistema realiza uma verificação adicional para localizar os processos que utilizem o mesmo processo como referência. Quando encontrado, o robô aplica a Etiqueta 5.

4. **Habeas Corpus:** Esses processos são analisados exclusivamente quando se encontram no 2º Grau do PJe. Esse item constitui um dispositivo processual de garantia fundamental previsto na Constituição Federal (art. 5º, inciso LXVIII, no Brasil), destinado a assegurar o direito à liberdade e locomoção. Neste cenário, o robô realiza a busca por prevenção identificando processos com a mesma referência. Se não encontrar nenhum, o sistema procura processos protocolados pelo mesmo advogado e já julgados anteriormente. Caso um número de referência relacionado seja encontrado no conteúdo das petições iniciais, o robô aplica a Etiqueta 5 ao primeiro processo prevento e a Etiqueta 2 aos demais.

Embora buscas estruturadas e o uso de expressões regulares sejam funcionais na identificação de possíveis casos de litigância predatória, sua eficácia pode ser limitada

em contextos onde há manipulação textual (MAGALHÃES; SOUSA et al., 2024). A busca semântica, por outro lado, tende a ser mais eficiente nesses cenários, pois não se restringe à correspondência exata de palavras-chave, mas interpreta o significado e a intenção subjacente a cada consulta (DAVE; LAWRENCE; PENNOCK, 2003). Modelos contextuais, como os baseados em redes neurais profundas — que são modelos de aprendizado de máquina compostos por múltiplas camadas de neurônios artificiais capazes de aprender representações complexas (LECUN; BENGIO; HINTON, 2015) — permitem uma compreensão mais abrangente do corpus textual, capturando relações semânticas entre os termos e aumentando a precisão na recuperação da informação.

2.3 Processamento de Linguagem Natural

O PLN surge como uma solução promissora para a identificação de litigância predatória, uma vez que as petições, recursos e demais peças processuais são documentos textuais que contêm informações valiosas para a detecção de práticas abusivas. Por meio da análise semântica e da comparação de textos, o PLN permite identificar similaridades entre processos, detectar padrões de repetição e evidenciar práticas abusivas. Essa abordagem é especialmente relevante no contexto atual, em que o volume de processos exige soluções automatizadas para análise abundante de dados (FILHO; JUNQUILHO, 2018).

Nesse cenário, o PLN apresenta-se como uma alternativa para lidar com a complexidade e variabilidade da linguagem humana em documentos jurídicos (OLIVEIRA; NASCIMENTO, 2022). O PLN é um campo interdisciplinar que combina técnicas de ML, Ciência de Dados e IA para desenvolver sistemas capazes de compreender, interpretar e gerar linguagem natural de forma automatizada (HIRSCHBERG; MANNING, 2015). Surgido na década de 1950 como uma interseção entre IA e linguística, o PLN inicialmente distinguia-se da RI, focando-se na compreensão e manipulação da linguagem natural. No entanto, com o avanço das tecnologias e a crescente demanda por soluções integradas, o PLN assimilou conceitos de diversas áreas, incluindo a RI, ampliando seu escopo de aplicação e impulsionado avanços significativos em ambas as áreas (SCHÜTZE; MANNING; RAGHAVAN, 2008; CHOWDHARY KR1442, 2020). Essa convergência entre PLN e RI tem sido fundamental para o desenvolvimento de sistemas capazes de analisar grandes volumes de documentos textuais, como os processos judiciais, de forma eficiente e precisa.

As técnicas de PLN são amplamente utilizadas na aplicação de modelos de ML para a solução de problemas e construção de sistemas complexos, que podem ser alimentados por coleções de dados estruturados ou não (MANNING; RAGHAVAN; SCHüTZE, 2008). O PLN revela-se uma abordagem particularmente útil no desenvolvimento de sistemas para o judiciário, considerando que a principal composição dos dados desse domínio são seus documentos, o que caracteriza um *corpus* essencialmente textual (FILHO; JUNQUILHO,

2018). Geralmente, o fornecimento de dados textuais para técnicas de PLN em modelos de ML exige um tratamento adequado para que esses dados possam ser utilizados de forma eficaz. Entre as etapas mais importantes para a preparação dos dados está o pré-processamento, que consiste em um conjunto de algoritmos específicos visando e padronizar os dados e, assim, melhorar o desempenho dos modelos de ML em tarefas específicas (CIRQUEIRA et al., 2018). Nesse contexto, o pré-processamento desempenha um papel crucial na construção de soluções baseadas em PLN, sendo uma etapa preparatória essencial antes da aplicação de algoritmos destinados a tarefas mais especializadas, como a própria RI.

Com os avanços no PLN, uma das técnicas mais utilizadas na criação de RI é a representação vetorial de dados textuais. Essa abordagem possibilita a resolução de diversas tarefas baseadas em IA, como análise de sentimentos, geração de texto, modelagem de tópicos e conversão de texto em fala.

2.4 Representações Vetoriais de Palavras

A análise de documentos textuais exige que o computador compreenda o significado das palavras e suas relações semânticas. No entanto, os computadores não são capazes de processar linguagem natural de forma direta (LAKE; MURPHY, 2023). Dessa forma, é necessário converter palavras em representações numéricas que capturam o seu significado e contexto. Nesse cenário, as representações de palavras exercem um papel fundamental na criação de sistemas modernos que interagem com o usuário através da linguagem natural (ZHANG et al., 2023a). As estratégias de representação vetorial variam desde abordagens simples, como o *One-Hot Encoding* (BISHOP; NASRABADI, 2006), até modelos avançados de linguagem, como o *BERT* (KENTON; TOUTANOVA, 2019) e o *GPT* (RADFORD, 2018). Enquanto o *One-Hot Encoding* representa cada palavra como um vetor esparso e de alta dimensionalidade, os modelos baseados em *embeddings* contextuais, como o *BERT*, utilizam técnicas mais sofisticadas para capturar o significado das palavras em um espaço vetorial denso e de menor dimensionalidade (ILIĆ et al., 2018).

Dentre as técnicas responsáveis por criar representações vetoriais a partir de palavras num espaço multidimensional, vale citar as baseadas em *embeddings* contextuais (ILIĆ et al., 2018). Esta categoria de representação vetorial utiliza uma abordagem baseada em *Transformers*, que é uma arquitetura de redes neurais que propõe a utilização de múltiplos mecanismos de atenção e é capaz de capturar o contexto global no qual aquela palavra está inserida (VASWANI, 2017). Essa abordagem levou ao desenvolvimento dos Modelos de Linguagem Pré-Treinados (PLMs), como BERT (KENTON; TOUTANOVA, 2019), GPT (RADFORD, 2018), T5 (RAFFEL et al., 2020) e RoBERTa (LIU, 2019), que são baseados na arquitetura *Transformers* e treinados com uma quantidade massiva de textos utilizando

uma abordagem não supervisionada (KENTON; TOUTANOVA, 2019). Sua estratégia de treinamento resulta na criação de representações numéricas contextuais de palavras e frases, aumentando significativamente o número de parâmetros (BROWN et al., 2020). Um dos benefícios dos PLMs é sua capacidade de atuar como modelos de *few-shot learning*, ou seja, aprender com poucos exemplos (ZHANG et al., 2023b). Essa característica possibilita a criação de sistemas baseados em *in-context learning*, eliminando a necessidade de ajustes nos hiperparâmetros do modelo ou *fine-tuning* (MOSBACH et al., 2023).

2.4.1 Transformers

A arquitetura *Transformer*, proposta por Vaswani (2017), revolucionou o aprendizado profundo e possibilitou o desenvolvimento de modelos pré-treinados (PTMs) de alto desempenho em PLN (LIN et al., 2022). Este padrão arquitetural propõe a utilização do mecanismo de autoatenção (Self-Attention), superando as limitações de Redes Neurais Recorrentes (RNNs) e Redes Neurais Convolucionais (CNNs) no processamento de sequências, especialmente em tarefas que envolvem dependências de longo alcance (VASWANI, 2017). Além disso, a arquitetura introduz o conceito de *Multi-Head Self-Attention*, que permite ao modelo capturar relações entre diferentes partes da sequência de entrada de forma paralela e eficiente (VASWANI, 2017).

Na Figura 1, é apresentada a arquitetura do *Transformer*, destacando seus principais componentes, como as camadas de *Self-Attention*, *Multi-Head Attention*, e as conexões residuais. A arquitetura utiliza uma entrada composta por *embeddings* vetoriais das palavras ou *tokens*, processados por camadas empilhadas de *Self-Attention*. Segundo (LIN et al., 2022), o mecanismo de *Self-Attention* permite que o modelo compare cada palavra ou *token* com todos os outros na sequência, atribuindo pesos de atenção que refletem a relevância contextual entre eles. A camada *Multi-Head Attention* combina as saídas de múltiplas *Attention-Heads*, cada uma focando em diferentes aspectos da sequência de entrada, gerando melhores representações através da captura do contexto (VASWANI, 2017).

Essa arquitetura *Transformer* é composta por codificadores e decodificadores. Os codificadores processam a entrada, gerando representações vetoriais das palavras ou *tokens* por meio do mecanismo de *Self-Attention*, que captura nuances e relações contextuais. Já os decodificadores geram a sequência de saída palavra por palavra, utilizando o *Self-Attention* para analisar as saídas do codificador e focar nas partes mais relevantes para a geração de cada *token* (VASWANI, 2017).

Além das camadas de atenção, a arquitetura inclui uma camada *Feed Foward*, que realiza transformações não lineares nas representações geradas pelo *Self-Attention* (VASWANI, 2017). A utilização desta camada permite capturar padrões mais complexos e gerar respostas mais coerentes e expressivas.

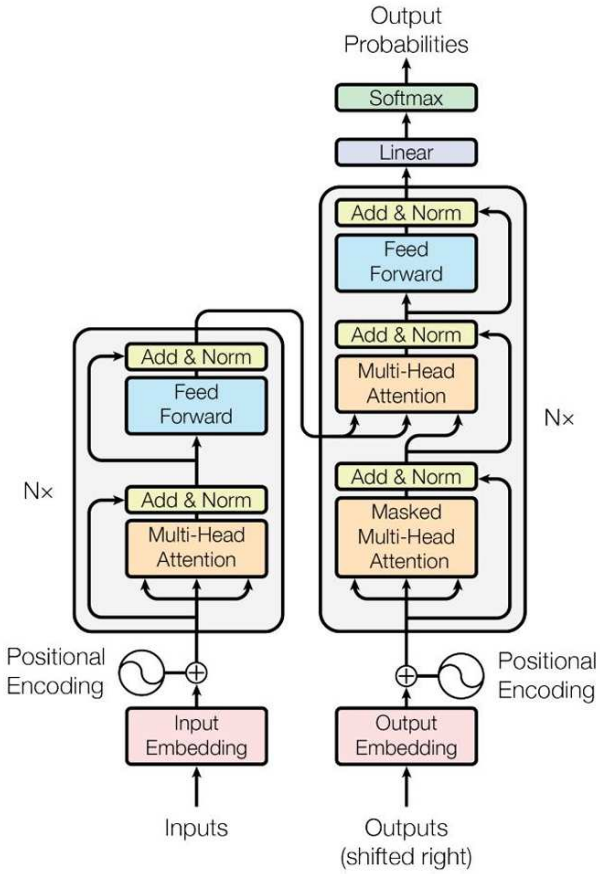

Figura 1 – Modelo Transformers
Fonte: Vaswani (2017)

A arquitetura também inclui camadas de normalização, que garante estabilização durante o treinamento e acelera a convergência do modelo. Por fim, as conexões residuais proporcionam o aprendizado de identidades, garantindo que o treinamento de modelos profundos seja uma tarefa mais eficiente.

Dentre os modelos desenvolvidos a partir da arquitetura *Transformers* existem os modelos BERT, que serão o foco desse estudo. A motivação para utilizar este modelo se dá pelo seu desempenho em tarefas de análise de similaridade semântica (POLO et al., 2021; BHATTACHARYA et al., 2022; OLIVEIRA; NASCIMENTO, 2022). Essa característica está alinhada ao objetivo deste trabalho, que busca avaliar a proximidade semântica entre diferentes textos e aprimorar a análise de litigância predatória. A Seção 2.4.2 detalha o funcionamento e os métodos de treinamento do BERT.

2.4.2 BERT

O BERT é um algoritmo de aprendizado profundo do *Google* para PLN. Ele foi o pioneiro na aplicação da técnica *Masked Language Model* (MLM) por utilizar uma abordagem bidirecional, que não existia em modelos anteriores (KENTON; TOUTANOVA,

2019). O modelo utiliza o *Transformer*, que possui camadas de atenção para identificar relações contextuais dentro do texto. Enquanto o *Transformer* inclui dois mecanismos divididos: codificador e decodificador, o BERT possui apenas um mecanismo codificador, visto que o seu objetivo é gerar um modelo de linguagem (RAFFEL et al., 2020).

O modelo possui o funcionamento diferente dos outros, através da utilização do codificador *Transformer*, ele realiza a leitura de todas as palavras de uma sequência de uma vez, enquanto os outros modelos leem as palavras da esquerda para a direita ou o contrário (GOLDBERG, 2017). Durante o treinamento de modelos de linguagem, um dos desafios recorrentes é realizar a tarefa de previsão das próximas sequências de palavras (BENGIO; DUCHARME; VINCENT, 2000). Para isso, o BERT utiliza duas estratégias de treinamento: o MLM e o *Next Sentence Prediction* (NSP) (KENTON; TOUTANOVA, 2019).

2.4.2.1 Previsão da Próxima Frase - NSP

No treinamento do BERT, o modelo é alimentado com pares de frases em sua entrada, que aprende a prever se a segunda frase do par corresponde à sequência correta no documento original (JURAFSKY; MARTIN, 2023). Ainda na fase de treinamento, metade das entradas são um par em que a segunda frase é a correta dentro do documento original, ao mesmo tempo que a outra metade possui frases aleatórias como par (JURAFSKY; MARTIN, 2023). A heurística seguida é a de que a sequência adicionada de forma aleatória será desassociada da primeira.

De acordo com (KENTON; TOUTANOVA, 2019), antes dos dados de entrada serem adicionados no modelo, eles são processados para que haja a distinção entre as sentenças. Dessa forma, um *token* de classificação - *Classification* (CLS) - é inserido na primeira frase e um *token* de separação - *Separation* (SEP) - é inserido no final de cada frase. Após isso, um *embedding* de frase indicando as frases são adicionadas a cada *token*. Esse tipo de *embedding* possuem um conceito semelhante aos *embeddings* de *token* com um vocabulário de dois. Por conseguinte, um *embedding* posicional é adicionado em cada *token* para indicar sua posição na sequência.

A abordagem seguida pelos autores (KENTON; TOUTANOVA, 2019) para verificar se a segunda sentença está conectada a primeira é a seguinte: (1) A sequência de entrada é submetida ao modelo do *Transformer*; (2) A saída do token CLS é convertida para um vetor na forma 2 x 1 fazendo uso de uma camada de classificação simples. (3) A probabilidade de *IsNextSequence* é calculada com a função de ativação *softmax*. (4) O treinamento do BERT é realizado em conjunto com o MLM e NSP, com o intuito de minimizar a função de perda combinada das duas estratégias.

2.4.2.2 Máscara de Palavras - MLM

Durante o treinamento do modelo BERT, são selecionados 15% dos *tokens* na entrada, os quais são escolhidos de forma aleatória (JURAFSKY; MARTIN, 2023). Esses *tokens* são pré-processados empregando a seguinte estratégia: 80% são substituídos por um *token* de mascaramento, 10% por uma palavra aleatória e 10% usam a palavra original (KENTON; TOUTANOVA, 2019).

Os autores (KENTON; TOUTANOVA, 2019) consideraram as seguintes suposições até chegarem na delimitação da estratégia de pré-processamento: (1) Caso utilizassem o mascaramento durante 100% do tempo, o modelo poderia não gerar boas representações de *tokens* para palavras não mascaradas. O modelo foi otimizado para fazer a previsão de palavras mascaradas, no entanto, os *tokens* não mascarados ainda poderiam ser utilizados para entendimento do contexto. (2) Caso utilizasse o mascaramento 90% das vezes e palavras aleatórias 10%, o modelo iria inferir que a palavra observada nunca estivesse correta; (3) Se fosse utilizado o mascaramento durante 90% do tempo e mantivessem a mesma palavra 10% do tempo, o modelo copiaria de maneira ordinária o *embedding* não contextual.

2.5 Recuperação de Informação

A Recuperação de Informação (RI) surgiu mediante a quantidade de dados que as bibliotecas armazenavam, assim, nos anos de 1960 foram propostas iniciativas para automatizar o armazenamento e a recuperação de informações bibliográficas (CASELI; NUNES, 2024). A RI possui como princípio encontrar documentos não estruturados armazenados em uma grande coleção de dados, satisfazendo a necessidade de informação (SCHÜTZE; MANNING; RAGHAVAN, 2008). O objetivo principal da RI é a busca, consistindo em encontrar o documento relevante para o usuário. Esta tarefa é conhecida como recuperação *ad hoc*, podendo ser aplicado em diferentes tipos de dados como imagem, vídeo e áudios (CASELI; NUNES, 2024).

Ainda segundo (CASELI; NUNES, 2024), a tarefa central da RI é encontrar documentos relevantes que correspondam à consulta do usuário. No entanto, uma das dificuldades dessa tarefa é a incompatibilidade de vocabulário (*vocabulary mismatch*), que ocorre quando os termos usados na consulta não estão presentes nos documentos. Os sistemas baseados em RI são úteis para tarefas de PLN, como sistemas de perguntas e respostas e detecção de plágio. Entre suas vantagens, destaca-se o baixo custo computacional, o que permite sua aplicação em tarefas mais complexas dentro de um sistema.

Na Figura 2, é apresentado uma visão geral de um sistema de RI. O ponto inicial do sistema consiste em um usuário, e o final corresponde a uma lista com os resultados

ordenados por meio de um *ranking*. A Etapa 1 e 2, representam a preparação do sistema e são feitas antes do modelo estar em produção. Na Etapa 1, os documentos são pré-processados através de tarefas como organização, limpeza textual e transformação textual. Na Etapa 2, é realizada a indexação desses dados, que pode ser realizado através da criação de um índice chamado de arquivo invertido, ele compõe um dicionário e uma lista de identificadores, armazenando também o número de documentos (*document frequency* - IDF). Na Etapa 3, o sistema executa a busca implementando as mesmas tarefas de pré-processamento da Etapa 1. O texto pré-processado é utilizado na Etapa 4 para buscar no índice dos documentos mais adequados para a consulta. A geração do *ranking* na Etapa 4, resulta em uma lista ordenada com os itens mais relevantes para aquela consulta.

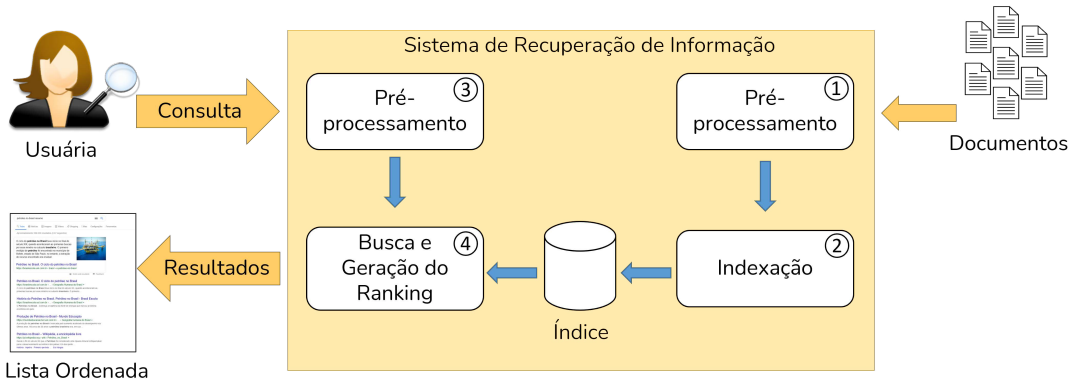

Figura 2 – Visão Geral de um Sistema de RI.
Fonte: (CASELI; NUNES, 2024)

2.5.1 Cálculo de Similaridade do Cosseno

A similaridade do cosseno é uma métrica implementada em trabalhos relacionados a recuperação de informação (RAHUTOMO et al., 2012). Esta métrica é utilizada para calcular a similaridade entre vetores de documentos (LAHITANI; PERMANASARI; SETIAWAN, 2016). Através deste cálculo é possível realizar buscas em grandes bases de dados com o conteúdo vetorizado, retornando os documentos mais relevantes para uma determinada consulta (SALTON; BUCKLEY, 1988).

Este método baseia-se no princípio de que o cosseno entre o vetor da consulta e o vetor do documento representa a similaridade entre eles. A Equação 2.1 calcula a similaridade do cosseno, definida como o produto escalar normalizado dos vetores $\vec{q}$ (consulta) e $\overrightarrow{d_j}$ (documento). O cosseno é máximo se os vetores possuem um ângulo de 0 graus, e o mínimo se os vetores formarem um ângulo de 90 graus. Caso o valor do cosseno for mínimo, os vetores não compartilham nenhum termo (CASELI; NUNES, 2024).

$$\text{cosseno}(q, d_j) = \frac{\vec{q} \cdot \vec{d_j}}{|\vec{q}| \times |\vec{d_j}|} \tag{2.1}$$

2.5.2 Medidas de Desempenho

Existem métricas específicas para sistemas baseados em RI, que utilizam o conceito de relevância do documento retornado. A relevância pode ser tratada de forma binária, onde só existem dois valores possíveis para ela, o relevante (1) e o irrelevante (0) (CASELI; NUNES, 2024). No entanto, em alguns contextos, a relevância pode ser graduada, como em escalas de 1 a 5, o que é particularmente útil para métricas como o *Normalized Discounted Cumulative Gain* (NDCG). As métricas possuem valores que variam de 0 a 1, onde 1 representa a recuperação ideal e 0 um item irrelevante para a pesquisa. As medidas de desempenho podem ser guiadas através de um *ground truth*, que estabelece a relação entre uma consulta e seus documentos relevantes esperados. As medidas de desempenho para SRIs podem ser divididas em duas: métricas baseadas em conjuntos e métricas para *rankings*.

2.5.2.1 Métricas Baseadas em Conjuntos

Na Figura 3, é representado os conjuntos que originam as métricas de avaliação em um sistema baseado em RI, onde é demonstrado que uma parte dos documentos relevantes e irrelevantes são recuperados durante uma operação no sistema. Com base nesses conjuntos é possível definir duas métricas para avaliar a qualidade da recuperação: a precisão (2.2) e a revocação (2.3).

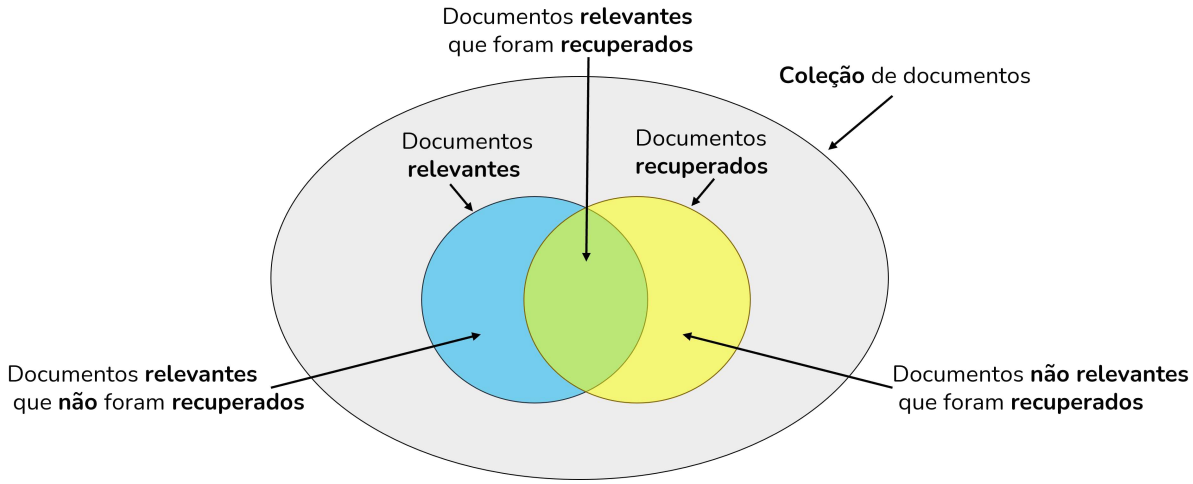

Figura 3 – Representação dos conjuntos de um sistema de RI
Fonte: Caseli e Nunes (2024)

$$\text{Precisão} = \frac{\#\text{relevantes recuperados}}{\#recuperados} \tag{2.2}$$

$$\text{Revocação} = \frac{\#\text{relevantes recuperados}}{\#relevantes} \tag{2.3}$$

Na realização do cálculo das medidas de desempenho, é comum notar que maiores níveis de *recall* acompanham baixos níveis de precisão, revelando uma relação inversamente proporcional entre as métricas. Isso ocorre porque, ao aumentar o número de documentos recuperados, a revocação tende a aumentar, mas a precisão pode diminuir, já que mais documentos irrelevantes podem ser incluídos. Dessa forma, pode ser adotada outra métrica que estabelece uma relação entre as medidas de precisão e revocação, a medida F, que pode ser representada pela Equação 2.4. Nesta equação, o valor de β é um parâmetro que permite o ajuste para ênfase em uma das métricas. Para definir que a ênfase seja as mesmas para as duas métricas, basta utilizar o valor de $\beta = 1$. Nesse caso, a métrica é comumente chamada de F1, e sua fórmula pode ser simplificada para a Equação 2.5.

$$F = \frac{(\beta^2 + 1) \cdot P \cdot R}{(\beta^2 \cdot P) + R} \tag{2.4}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{2.5}$$

2.5.2.2 Métricas para *Rankings*

Os sistemas de RI retornam os documentos em *rankings*. Uma característica fundamental dos *rankings* é a ordenação dos resultados, onde a posição de cada documento é importante. Diferentemente das métricas baseadas em conjuntos, que tratam os resultados como grupos não ordenados, as métricas para *rankings* consideram a posição dos documentos relevantes.

Por exemplo, considere dois sistemas A e B que retornam os mesmos 5 documentos, dos quais 3 são relevantes. No sistema A, os documentos relevantes estão nas posições 1, 2 e 3, enquanto no sistema B estão nas posições 3, 4 e 5. Usando métricas baseadas em conjuntos, ambos teriam a mesma precisão (0.6) e revocação (1.0), mas é evidente que o sistema A possui um desempenho superior, pois retorna os documentos relevantes logo no topo.

Para considerar a ordenação dos documentos, existem métricas específicas que consideram tanto a relevância quanto a posição dos documentos, como *Precision@k* (Equação 2.6) e *Recall@k* (Equação 2.7), que avaliam os k primeiros resultados.

$$\textit{Precision@k} = \frac{\text{Número de documentos relevantes nos primeiros k resultados}}{k} \tag{2.6}$$

A métrica *Precision@k* mede a proporção de documentos relevantes entre os k primeiros retornados pelo sistema. Essa métrica é útil quando há um limite fixo no número de resultados exibidos, como em mecanismos de busca ou sistemas de recomendação. No

entanto, a *Precision@k* não leva em conta a quantidade total de documentos relevantes disponíveis no conjunto.

$$\textit{Recall@k} = \frac{\text{Número de documentos relevantes nos primeiros k resultados}}{\text{Número total de documentos relevantes no conjunto}} \tag{2.7}$$

A métrica *Recall@k*, por outro lado, mede a fração de documentos relevantes recuperados dentro dos k primeiros resultados em relação ao número total de documentos relevantes disponíveis no conjunto. Isso significa que um sistema pode ter um *Recall@k* alto mesmo que a precisão seja baixa, caso ele consiga recuperar muitos documentos relevantes, independentemente da presença de itens irrelevantes na lista retornada.

Além disso, a métrica *Mean Average Precision* (MAP) (Equação 2.10), é a média dos valores de *Average Precision (AP)* (Equação 2.8) calculados para múltiplas consultas. A MAP sintetiza a relação precisão-revocação em um único valor entre 0 a 1, onde quanto mais próximo de 1 mais acurado é o modelo. Dessa forma, a AP considera tanto a quantidade de documentos relevantes quanto o quão próximos do topo do *ranking* eles estão. Para isso, calcula-se a média das precisões obtidas em cada posição onde um documento relevante é encontrado, penalizando documentos relevantes que aparecem em posições mais baixas no *ranking*.

$$AP = \frac{\sum_{k=1}^{n} P(k) \cdot rel(k)}{\#\text{relevantes}} \tag{2.8}$$

Onde $P(k)$ é a precisão calculada na posição k do *ranking* e $rel(k)$ é uma função binária que retorna 1 se o documento na posição k é relevante, e 0 caso contrário.

Por exemplo, Suponha que um sistema retorne os seguintes documentos (R = relevante, N = irrelevantes):

Posição k	Documento	Precisão até k	Relevância $rel(k)$
1	R	$1/1 = 1.0$	1
2	N	$1/2 = 0.5$	0
3	R	$2/3 = 0.67$	1
4	R	$3/4 = 0.75$	1

Tabela 2 – Cálculo de AP

O AP será:

$$AP = \frac{(1.0 + 0.67 + 0.75)}{3} = 0.807 \tag{2.9}$$

$$MAP = \frac{\sum_{q=1}^{|Q|} AP_q}{|Q|} \tag{2.10}$$

Por exemplo, se tivermos duas consultas com AP de 0.8 e 0.6, o MAP seria $(0.8 + 0.6)/2 = 0.7$.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \tag{2.11}$$

O *Mean Reciprocal Rank* (MRR) (Equação 2.11) é especialmente útil em cenários onde apenas a posição do primeiro documento relevante é importante, como em sistemas de busca de perguntas e respostas. No exemplo disposto na Tabela 3, se em três consultas o primeiro documento relevante aparece nas posições 3, 1 e 2, o MRR seria $(1/3 + 1/1 + 1/2)/3 = 0.61$.

Consulta	Primeira posição relevante	$\frac{1}{\text{posição}}$
Q1	3	$1/3 = 0.33$
Q2	1	$1/1 = 1.00$
Q3	2	$1/2 = 0.50$

Tabela 3 – Cálculo do MRR

$$MRR = \frac{0.33 + 1.00 + 0.50}{3} = 0.61 \tag{2.12}$$

Em alguns casos, é importante diferenciar um documento muito relevante de um parcialmente relevante. A técnica *Normalized Discounted Cumulative Gain* (NDCG) proposta por (JÄRVELIN; KEKÄLÄINEN, 2002) permite trabalhar com níveis graduados de relevância, por exemplo, 0 para irrelevante, 1 para parcialmente relevante e 2 para muito relevante, e considera a posição dos documentos no *ranking*.

O *Discounted Cumulative Gain* (DCG) é calculado da seguinte forma:

$$DCG = \sum_{i=1}^{n} \frac{2^{rel_i} - 1}{\log_2(i+1)} \tag{2.13}$$

Para normalizar o DCG, garantindo que o valor fique entre 0 e 1, calcula-se o *Ideal Discounted Cumulative Gain* (IDCG), que representa o melhor *ranking* possível (ordenando os documentos por relevância perfeita). O NDCG é então calculado como:

$$NDCG = \frac{DCG}{IDCG} \tag{2.14}$$

Por exemplo, considere um *ranking* com 3 documentos onde $rel = [2, 1, 2]$. O DCG seria calculado como:

$$DCG = \frac{2^2 - 1}{\log_2(2)} + \frac{2^1 - 1}{\log_2(3)} + \frac{2^2 - 1}{\log_2(4)} = 4.13 \tag{2.15}$$

A seguir, a Tabela 4 apresenta um exemplo de cálculo da NDCG para um conjunto de documentos ordenados por relevância:

Posição	Documento	Relevância *rel*	DCG Parcial	IDCG Parcial
1	Doc A	2	3.00	3.00
2	Doc B	1	0.63	1.89
3	Doc C	2	1.50	1.50
Total			4.13	6.39

Tabela 4 – Exemplo de cálculo do NDCG

O NDCG é então calculado como:

$$NDCG = \frac{4.13}{6.39} = 0.65 \tag{2.16}$$

Dessa forma, se todos os documentos relevantes aparecem nas primeiras posições, o NDCG será próximo de 1. Do contrário, se documentos relevantes estão mal posicionados, o NDCG será menor.

Essas métricas são amplamente utilizadas em contextos de RI. Embora outras métricas sejam debatidas em fóruns especializados como o *Special Interest Group on Information Retrieval* (SIGIR) e o *Text Retrieval Conference* (TREC), elas não têm ampla adoção. A principal vantagem de utilizar as métricas expostas reside nos estudos já consolidados, como o realizado por Buckley e Voorhees (2017), que demonstrou sua eficácia em avaliar sistemas de recuperação de informação em diferentes contextos e aplicações.

3 TRABALHOS RELACIONADOS

O aumento no volume de dados no judiciário mediante ao processo de modernização exige que novas tecnologias sejam empregadas para dar eficiência na aplicação da lei. Nesse sentido, existem vários estudos que aplicam técnicas de PLN para solucionar diversos problemas da indústria, inclusive no Judiciário. Esta seção apresenta os trabalhos relacionados que serviram de apoio para o desenvolvimento desta pesquisa.

3.1 Aplicação de *Embeddings* no Domínio Jurídico

A relevância da aplicação de técnicas de PLN no jurídico foi discutida por Carmo (2024), que utilizou modelos de linguagem baseados em *embeddings* contextuais para a identificação de precedentes através de algoritmos de classificação. Neste trabalho, é ressaltado que o uso de modelos especializados podem gerar representações vetoriais significativas para tarefas de PLN, pois permitem capturar as nuances do vocabulário específico do domínio, aumentando a eficiência em tarefas mais direcionadas.

Outro exemplo de pesquisa relacionada a solução de problemas do Judiciário por meio de *embeddings* foi feita por Mentzingen et al. (2024), o qual propôs a utilização de técnicas baseadas em PLN para identificar casos semelhantes. A pesquisa relata que modelos que geram representações vetoriais baseado em palavras podem ser empregados na tarefa de identificação de precedentes. Outro ponto relevante destacado pelo trabalho, é que o refinamento de modelos garantem melhores resultados em tarefas de classificação. Outrossim, os experimentos realizados promovem avanços em sistemas de decisão no contexto judicial, auxiliando na tomada de decisão quanto na condução de novos processos.

De forma análoga, o trabalho de Polo et al. (2021) propõe a criação de modelos de linguagem pré-treinados para a linguagem jurídica brasileira. A pesquisa surge a partir da dificuldade do tratamento da informação que circula dentro dos tribunais de justiça, os quais exigem uma linguagem formal específica, revelando uma lacuna em recursos para automatizar processos baseados em dados jurídicos. Este trabalho faz uma avaliação de *embeddings* contextuais através de algoritmos de classificação. Concluindo que modelos de linguagem treinados para dados jurídicos possuem desempenho melhor que generalistas para aplicações neste domínio.

3.2 Cálculo de Similaridade Semântica

A pesquisa de Petrović e Stanković (2019) investigou o impacto de diferentes métodos de pré-processamento na similaridade textual, demonstrando que a escolha dessas técnicas pode afetar significativamente os resultados da análise semântica. O estudo avaliou ferramentas de normalização, remoção de *stopwords* e lematização, identificando que a aplicação dessas técnicas resultou em uma melhora na recuperação da informação e na precisão da análise semântica. Essa análise reforça a importância de uma etapa criteriosa de pré-processamento para garantir representações vetoriais mais eficazes. Esse resultado se alinha ao presente trabalho, que também adota técnicas de normalização e conversão textual para otimizar a similaridade semântica entre processos judiciais.

O estudo de Oliveira e Nascimento (2021) propõe aplicar técnicas de similaridade em documentos jurídicos para auxiliar na busca por jurisprudência. Este trabalho dispõe de uma abordagem não supervisionada baseada na aplicação de *embeddings* em uma base de dados de Recursos Ordinários Interpostos (ROI) na tarefa de clusterização. Além disso, destaca-se que modelos especializados baseados na incorporação de palavras possuem melhores desempenhos para tarefas de um domínio específico. O trabalho demonstra que o cálculo de similaridade possui um papel importante na identificação de precedentes para julgamento consistente de novos processos no judiciário.

A pesquisa de Oliveira e Nascimento (2022) explora o uso de modelos de linguagem baseados em transformadores, como *Bidirectional Encoder Representation from Transformers* (*BERT*), *Generative Pre-Trained Transformer-2* (GPT-2) e *Robustly Optimized BERT Approach* (*RoBERTa*), adaptados ao contexto jurídico. Esses modelos foram treinados com um vasto conjunto de documentos judiciais para criar versões especializadas e mais alinhadas a esse domínio. A abordagem proposta empregou técnicas não supervisionadas para agrupar documentos com base em sua similaridade, o modelo *RoBERTa ptBR* se destacou alcançando melhor desempenho na tarefa. Curiosamente, esse modelo generalista superou os especializados, desafiando expectativas anteriores na literatura ao combinar eficiência computacional e qualidade nos resultados.

3.3 Recuperação de Informação

A recuperação de informações por meio de Sistemas de Recuperação da Informação (SRI) pode ser útil em cenários em que o usuário precisa encontrar documentos sobre determinado assunto. Diante disso, um estudo realizado por Costa (2024) propõe uma abordagem para a recuperação de jurisprudências em processos do Tribunal de Contas da União (TCU). Essa abordagem consiste em identificar e localizar decisões anteriores relevantes com base em critérios de similaridade, facilitando a análise de casos semelhantes e o embasamento de novas decisões. No estudo, foi demonstrado que representações vetoriais

do tipo *bag-of-words* (BoW) — uma técnica que cria representações vetoriais de textos, transformando-os em vetores de características que consideram apenas a frequência de ocorrência das palavras — podem ser úteis para sistemas de recuperação baseados em algoritmos, como o cálculo da similaridade do cosseno e o BM25. Além disso, o trabalho sugeriu a criação de uma nova abordagem que combina representações conceituais da base de dados com o *thesaurus* do TCU. O *thesaurus* é um dicionário que apresenta os significados de termos específicos de um determinado domínio. A abordagem proposta demonstra que a priorização de palavras com maior relevância para o contexto, na construção das representações vetoriais, pode trazer melhorias significativas em comparação com estratégias convencionais de RI.

Outro trabalho que trata de recuperação/recomendação de informação é proposto por Bhattacharya et al. (2022). O estudo combina similaridade textual e análise de redes de citações, incluindo referências a estatutos, para melhorar a precisão na identificação de casos similares. A validação do modelo foi realizada com uma base *ground truth*, criada a partir da avaliação de especialistas em direito, técnica também adotada no presente trabalho. Os resultados mostraram que a combinação de abordagens textuais e baseadas em rede superou métodos tradicionais, aumentando a correlação com a opinião dos especialistas em até 20,6%. Essa abordagem destaca a importância de integrar múltiplas fontes de informação para melhorar a recuperação de casos jurídicos.

3.4 Conclusões acerca dos trabalhos correlatos

A análise dos trabalhos mencionados destaca avanços expressivos no uso de técnicas de PLN e *Machine Learning* (ML) para a identificação e classificação de documentos no setor jurídico. Um panorama geral desses resultados pode ser observado na Tabela 5, que resume os trabalhos analisados, incluindo exemplos de modelos, métodos, técnicas e considerações que serviram de base para a condução deste trabalho.

O trabalho de Carmo (2024) ressalta que modelos especializados no *corpus* de um domínio específico, como o *BumbaBERT SC*, apresentam resultados superiores aos generalistas em tarefas como classificação de documentos jurídicos. Além disso, Mentzingen et al. (2024) reforça que representações vetoriais, como as geradas pelo *Word2Vec*, são eficazes na busca por precedentes, auxiliando magistrados na tomada de decisão.

Outro estudo relevante, proposto por Oliveira e Nascimento (2022), demonstra como o cálculo de similaridade de representações vetoriais pode contribuir para a criação de ferramentas aplicadas ao judiciário, destacando o uso de modelos como o *RoBERTa ptBR* e o *BERT ptBR*. Esses trabalhos evidenciam a importância de avaliar abordagens generalistas e especializadas para melhorar a precisão e a eficiência na recuperação de informações jurídicas.

Nesse contexto, este trabalho propõe uma análise comparativa de representações vetoriais geradas por modelos generalistas, como o *RoBERTa ptBR* e o *BERTimbau*, e modelos especializados, como o *BumbaBERT Small* e o *BERTikal*, na tarefa de recuperação de informação para recomendação de casos de litigiosidade predatória. O cerne da pesquisa, alinhado aos objetivos do programa Justiça 4.0, visa oferecer novas alternativas para a criação de ferramentas de IA no direito, promovendo celeridade no fluxo operacional do sistema judiciário brasileiro.

Autor/Ano	**Aplicação**	**Modelo/Método/Técnica**	**Considerações**
Carmo (2024)	Classificação de Precedentes	Modelos de *embeddings* pré-treinados baseados em BERT (ex. *BumbaBERT SC*).	Modelos especializados no domínio jurídico são superiores aos generalistas; classificação de documentos jurídicos.
Mentzingen et al. (2024)	Identificação de Precedentes Jurídicos Similares	Criação de *embeddings* baseados em transformadores e *bag-of-words* (*TF-IDF*, *Word2Vec*, *Doc2Vec*, *LDA* e *BERT*).	O melhor modelo utilizado foi o *Word2Vec*, demonstrando que representações granulares são eficazes na identificação de precedentes legais relevantes.
Bhattacharya et al. (2022)	Sistema de Recuperação da Informação.	Abordagem de similaridade e rede de citações utilização representações vetoriais *Word2Vec*, *BERT*, *LegalBERT*, *BERT-PLI*, *RoBERTa*.	Modelos como *Word2Vec* são eficazes na identificação de casos semelhantes, podendo ser comparado com a avaliação humana.
Petrović e Stanković (2019)	Impacto do Pré-Processamento na Similaridade Textual	Avaliação comparativa de diferentes métodos de pré-processamento para análise de representações vetoriais na similaridade textual.	A escolha das técnicas de pré-processamento impacta a análise semântica, podendo melhorar ou distorcer a recuperação de informação.
Oliveira e Nascimento (2022)	Agrupamento de Documentos Jurídicos	Utilização de *embeddings* baseados no *BERT* (*RoBERTa ptBR* e *BERT ptBR*), e no *GPT* (*GPT-2 ptBR*).	O modelo generalista *RoBERTa ptBR* atingiu os melhores resultados no agrupamento e melhor desempenho computacional.
Polo et al. (2021)	Classificação de Documentos Jurídicos	Modelos de *embeddings* pré-treinados: *Word2Vec*, *Doc2Vec*, *BERT-Base*, *BERTimbau*, *BERTikal*.	Modelos especializados no domínio jurídico são superiores aos generalistas; Classificação de documentos jurídicos.
Costa (2024)	Sistema de Recuperação da Informação.	Utilização de representações vetoriais (*BoW*, *TF-IDF*, *BM25*, *BoC*, *BoC-Th*) para aplicação jurisprudencial no TCU.	A nova abordagem proposta na criação de representações vetoriais BoC-Th atingiu valores satisfatórios.
Proposta	Recuperação de Informação	Busca por similaridade semântica utilizando modelos de linguagem baseados no *BERT* e RI para recomendação de possíveis casos de litgiosidade predatória.	Qual a representação baseada em *embeddings* é mais adequada para recomendar processos que praticam litigância predatória?

Tabela 5 – Resumo dos trabalhos correlatos analisados.
Fonte: Elaborado pelo autor (2024)

4 MATERIAIS E MÉTODOS

Neste Capítulo será discorrido sobre a metodologia utilizada para alcançar os objetivos desta pesquisa. Além disso, será justificada a utilização do método adotado e o detalhamento através da criação das instâncias inspiradas nas etapas propostas pela metodologia.

4.1 Metodologia

Este estudo propõe o desenvolvimento de um *framework* baseado em RI para a identificação de casos de litigiosidade predatória. O desenvolvimento será baseado nas etapas definidas pela metodologia *CRoss Industry Standard Processing for Data Mining* (CRISP-DM). Este método propõe uma abordagem interativa, permitindo que etapas anteriores possam ser revisitadas, visando aperfeiçoamento da modelagem proposta (RADFORD, 2018). O CRISP-DM possui ampla aplicação em trabalhos de ciências de dados e aprendizado de máquina devido a sua estrutura flexível, podendo ser aplicado em diversos setores, facilitando a interpretação do problema a partir de dados (MARTÍNEZ-PLUMED et al., 2019).

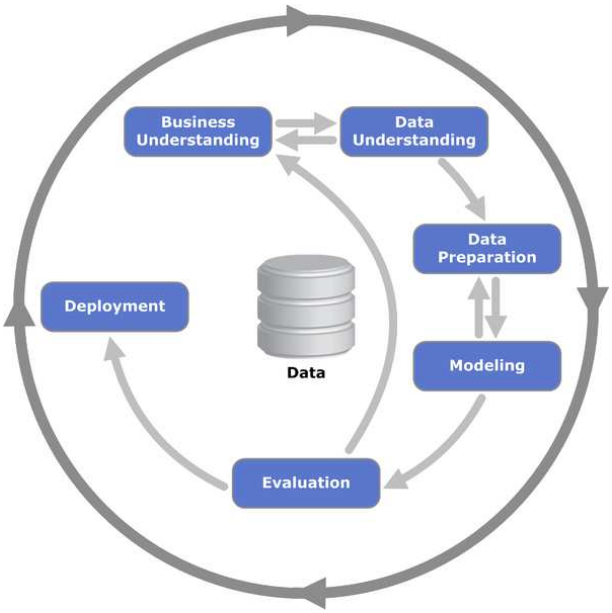

Figura 4 – Metodologia CRISP-DM
Fonte: (WIRTH; HIPP, 2000)

A Figura 4 destaca as seis fases do CRISP-DM, que são: (1) Entendimento do Negócio; (2) Entendimento dos Dados; (3) Preparação dos Dados; (4) Modelagem; (5) Avaliação; e (6) Implantação. As atividades realizadas em cada etapa são:

1. **Entendimento do Negócio:** Definição dos objetivos e requisitos comerciais dos projetos alinhados a análise de dados;
2. **Entendimento dos Dados:** Realização de coleta, exploração e análise dos dados disponíveis para avaliar a qualidade e identificação de padrões iniciais;
3. **Preparação dos Dados:** Seleção, limpeza e transformação dos dados, garantindo que estão prontos para uso;
4. **Modelagem:** Aplicação de algoritmos de aprendizado de máquina ou técnicas estatísticas para criação de modelos preditivos, ou descritivos;
5. **Avaliação:** Utilização de métricas nos resultados do modelo para verificar se os objetivos iniciais foram atingidos, determinando a validade e utilidade da modelagem;
6. **Entrega:** Obtenção dos resultados e compartilhamento de *insights* para atender às necessidades práticas do negócio.

A estrutura do CRISP-DM foi utilizada para organizar e guiar as etapas deste trabalho, permitindo flexibilidade e revisão iterativa.

4.1.1 Entendimento do Problema

Na primeira etapa do desenvolvimento do *framework*, foi realizado um estudo sobre como as demandas de litigiosidade predatória impactam o fluxo operacional do judiciário. Nesse sentido, o robô Nirie é responsável por identificar esses casos a partir da aplicação de rótulos, conforme ilustrado na Figura 5. O funcionamento do robô foi detalhado na Seção 2.2, que consiste em etapas que antecedem a tramitação do processo no PJe, a saber, (1) Envio do processo ao PJe; (2) Busca por processos preventos; (3) Extração do texto do documento; (4) Identificação da petição inicial; (5) Aplicação de etiquetas.

Conforme o departamento de TI do TJMA o modelo de operação do robô na identificação de litigiosidade predatória é uma abordagem que possui bons resultados. No entanto, essa abordagem pode gerar falsos positivos, uma vez que processos distintos podem compartilhar partes e advogados sem configurarem litigância predatória. Dessa forma, empregar técnicas de similaridade semântica podem aumentar a precisão na identificação de casos de litigiosidade predatória, justificando a proposta deste trabalho.

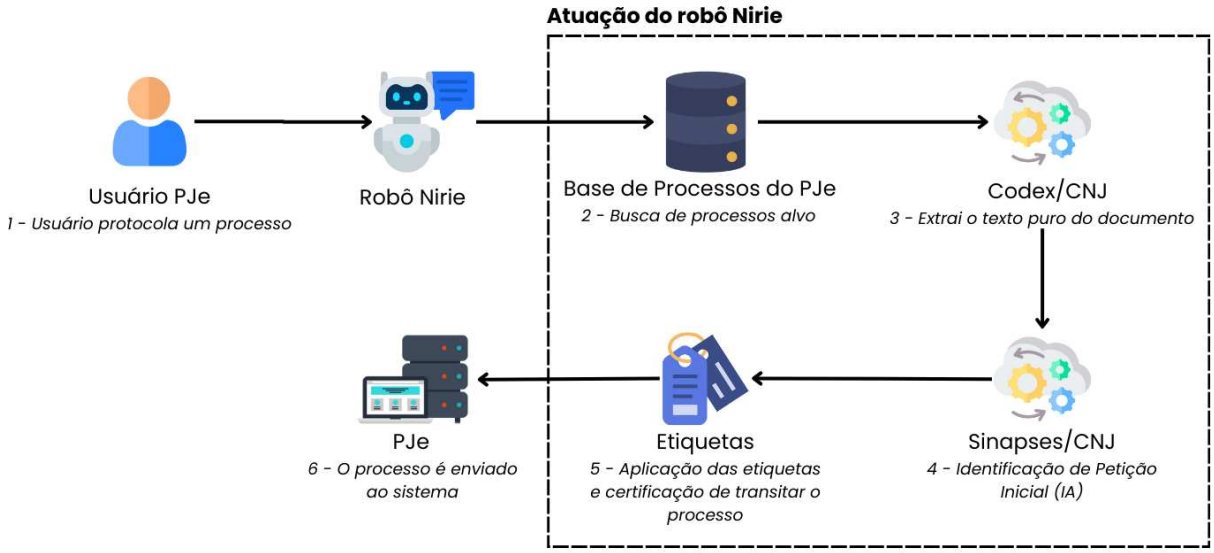

Figura 5 – Fluxo de funcionamento do robô Nirie
Fonte: Adaptado do relatório do Departamento de TI do TJMA (2024)

4.1.2 Entendimento dos Dados

A base de dados foi obtida por meio do Acordo de Cooperação n.º 002/2021, no qual o Departamento de TI do TJMA disponibilizou um conjunto com 36.820 processos judiciais relacionados a empréstimos consignados, no formato *Javascript Object Notation* (JSON). Na Tabela 6 são dispostos os detalhes da base de dados cedida.

A Tabela 6 apresenta a descrição da base de dados fornecida pelo Departamento de TI do TJMA, composta por 14 colunas que contêm informações relevantes para a protocolação de processos. Entre as colunas de interesse para este trabalho destacam-se "str_partes_polo_ativo", "str_partes_polo_passivo" e "str_nm_advogado", pois são utilizadas nas buscas estruturadas do robô Nirie. Além disso, o campo "texto_peticao", empregado pelo robô para buscas de expressões regulares, será utilizado neste trabalho para análise de similaridade semântica. Por fim, a coluna "str_nm_etiqueta" com as etiquetas aplicadas aos processos, conforme descrito na Seção 2.1.

Índice	Nome da Coluna	Tipo	Descrição
0	_id	object	Identificador único dos registros na base de dados.
1	nr_processo	object	Identificadores dos processos no sistema PJe.
2	dt_distribuicao	object	Data em que o processo foi protocolado.
3	ds_orgao_julgador	object	Nome do órgão julgador vinculado ao Programa Justiça 4.0.
4	cd_classe_judicial	int64	Código da classe judicial, que classifica os processos conforme sua natureza e objetivo.
5	ds_classe_judicial	object	Nome da classe judicial, que categoriza os processos conforme sua natureza e objetivo.
6	str_nm_etiqueta	object	Nome das etiquetas aplicadas pelo robô Nirie.
7	int_id_advogado	int64	Identificador único dos advogados no sistema.
8	str_nm_advogado	object	Nome do advogado vinculado ao processo.
9	str_oab	object	Número de inscrição do advogado na Ordem dos Advogados do Brasil (OAB).
10	id_peticao	int64	Identificador único da petição no sistema.
11	texto_peticao	object	Conteúdo textual da petição inicial do processo.
12	str_partes_polo_ativo	object	Nome do autor do processo.
13	str_partes_polo_passivo	object	Nome do réu (pessoa física ou jurídica) citado no processo.

Tabela 6 – Descrição da base de dados disponibilizada pelo Departamento de TI do TJMA. Fonte: Elaborado pelo autor (2025).

Na Tabela 7 é apresentada a distribuição das etiquetas aplicadas aos processos. Observa-se que a base de dados contém apenas 34 exemplos de processos classificados como possíveis demandas predatórias (Etiqueta 4). Além disso, a base disponibilizada não possui a pontuação de similaridade ou a discriminação dos pares de documentos relevantes entre si, tornando inviável a realização da avaliação do *framework* proposto, conforme descrito na Seção 4.1.6, uma vez que as técnicas de avaliação de *SRIs* exigem um *ground truth* relacionando pares de documentos para medir o desempenho do sistema de recuperação (COSTA, 2024).

Etiqueta	**Quantidade**
SEM ETIQUETA	36672
Etiqueta 2	68
Etiqueta 3	46
Etiqueta 4	34

Tabela 7 – Distribuição de etiquetas do conjunto disponibilizado pela equipe de TI do TJMA.

Fonte: Elaborado pelo Autor (2025).

O *ground truth* consiste em um conjunto de dados de referência que permite validar os resultados gerados pelo *framework* com base em uma correspondência esperada (MANNING; RAGHAVAN; SCHüTZE, 2008). No contexto deste estudo, o ideal seria uma base que inclua pares de documentos previamente classificados como similares. No entanto, a análise da base fornecida revelou a ausência desses pares, impossibilitando uma avaliação quantitativa confiável da modelagem proposta. Dessa forma, as limitações da base disponibilizada inviabilizam sua utilização para este estudo. Para viabilizar o processo avaliativo, foi necessária a adoção de um novo conjunto de dados que atendesse a esses critérios.

Diante disso, será utilizada a base de dados fornecida no estudo de Bhattacharya et al. (2022), que disponibiliza um conjunto estruturado de pares de documentos legais, como jurisprudências e estatutos. Essa base contém um *ground truth*, permitindo a validação do *framework* proposto. O *ground truth* consiste em uma lista de pares de documentos acompanhados de suas respectivas pontuações de similaridade, atribuídas por especialistas da área jurídica, conforme descrito na Seção 4.1.3.

Os dados utilizados por Bhattacharya et al. (2022) são provenientes da Suprema Corte da Índia (*SCI*) e incluem um acervo de 53.210 documentos de casos e 12.814 atos do judiciário indiano. A partir desse material, os autores estruturaram dois conjuntos de dados distintos: um conjunto de validação, contendo 143 documentos, e um conjunto de testes, composto por 153 documentos. O conjunto de validação é uma base de dados utilizada para ajustar e validar os hiperparâmetros do modelo. Já o conjunto de teste é empregado na avaliação final, permitindo verificar o desempenho do modelo de forma independente após a etapa de validação.

Categoria	Tipo	Descrição
Pasta de Documentos	*plain text*	Conjunto de arquivos contendo decisões judiciais e estatutos.
Pontuações de Similaridade	*CSV*	Lista de pares de documentos com suas respectivas pontuações de similaridade.

Tabela 8 – Base de dados disponibilizada por Bhattacharya et al. (2022)
Fonte: Elaborado pelo autor (2025).

Na Tabela 8 é apresentado um resumo da estrutura do conjunto de dados disponíveis. A base é composta por uma pasta contendo documentos jurídicos no formato de texto, escritos na língua inglesa. Além disso, o *ground truth* é disponibilizado como uma lista estruturada de pares de documentos, acompanhados de suas respectivas pontuações de similaridade. A partir dessas pontuações, foi possível realizar a avaliação da eficácia dos modelos utilizados na tarefa de RI.

4.1.3 *Ground Truth*

De acordo com Manning, Raghavan e Schütze (2008) um *ground truth* deve ser construído por especialistas do domínio ou por sistemas altamente confiáveis, garantindo a validade dos dados. No contexto de SRIs o *ground truth* consiste em documentos rotulados como "relevantes" ou "irrelevantes" em relação a uma consulta específica.

No trabalho de Bhattacharya et al. (2022) o processo de anotação foi realizado por especialistas em direito de duas universidades indianas. A base de teste, composta por 90 pares de documentos, foi anotada por dois especialistas da Universidade Nacional de Ciências Jurídicas de Bengala Ocidental (WBNUJS). Já a base de validação contou com a participação de três especialistas da Escola Rajiv Gandhi de Direito da Propriedade Intelectual (RGSOIPL). A pontuação de similaridade para cada par de documentos foi atribuída em uma escala de 0 a 1, sendo que a pontuação final resultou da média das anotações feitas pelos especialistas.

A colaboração entre especialistas de diferentes instituições possibilitou a criação de uma base de dados capaz de generalizar, de forma satisfatória, a avaliação da similaridade em processos. Esse trabalho resultou em uma lista de 100 pares de documentos, cada um com sua respectiva pontuação de similaridade. No processo de anotação foi utilizado o *Inter-Annotator Agreement* (IAA), medido através da métrica de correlação de *Pearson*. Assim, os pares de documentos foram segmentados em três categorias de similaridade.

- **0 a 0,4:** Documentos diferentes;

- **0,4 a 0,7:** Documentos moderadamente semelhantes;
- **0,7 a 1,0:** Documentos semelhantes (relevantes).

De acordo com a classificação proposta pelos autores, ao considerar os pares com pontuação de similaridade na faixa de [0.7, 1.0), identificam-se 30 pares de documentos relevantes entre si e 70 pares considerados irrelevantes, onde cada documento possui apenas um par relevante. Com isso, foi possível utilizar o *ground truth*, viabilizando a avaliação do *framework* nesse trabalho.

4.1.4 Preparação dos Dados

O processo de preparação dos dados foi dividido em duas fases principais: (i) tradução dos documentos para o português, a fim de garantir compatibilidade com os vetorizadores pré-treinados no idioma alvo; e (ii) pré-processamento do *corpus*, aplicando técnicas de normalização textual amplamente utilizadas em tarefas de mineração de textos.

Como os textos da base da SCI estão originalmente em inglês e os vetorizadores avaliados no *framework* proposto foram treinados com um *corpus* em português, foi necessário realizar a tradução desses documentos. Para esse processo, utilizou-se a *Application Programming Interface* (API) do *Google Translator*, adquirida por meio do serviço da *Google Cloud Platform* (GCP). Essa etapa foi essencial para garantir a compatibilidade entre os textos traduzidos e os modelos de linguagem aplicados na análise.

Dessa forma, os textos passaram por um pré-processamento utilizando a biblioteca Legal Text Preprocessing, desenvolvida no contexto do Acordo de Cooperação. Os métodos aplicados seguem padrões amplamente utilizados em trabalhos de mineração de textos, incluindo conversão para minúsculas, remoção de espaços em excesso, números, caracteres não alfanuméricos, acentuação e stopwords, além da limpeza geral do texto (CIRQUEIRA et al., 2018).

Em seguida, os textos passaram por um pré-processamento utilizando a biblioteca *Legal Text Preprocessing*, desenvolvida no contexto do Acordo de Cooperação. Os métodos aplicados seguem padrões amplamente utilizados em trabalhos de mineração de textos, incluindo conversão para minúsculas, remoção de espaços em excesso, números, caracteres não alfanuméricos, acentuação e *stopwords*, além da limpeza geral do texto (CIRQUEIRA et al., 2018). Estudos, como o de Petrović e Stanković (2019), demonstraram que a aplicação de técnicas como normalização, remoção de *stopwords* e lematização pode resultar em melhorias na recuperação de informação e na precisão da análise semântica. No entanto, o impacto dessas técnicas pode variar dependendo do domínio do texto e da tarefa específica.

Após essas etapas, foi gerada uma base de dados pré-processada, pronta para a extração das representações vetoriais que serão utilizadas na avaliação do *framework* proposto.

Para a análise dos processos judiciais, os documentos passaram por tradução automatizada para o português quando necessário. Devido à limitação de 512 *tokens* imposta pelos modelos adotados, apenas um segmento inicial de cada documento foi utilizado. Os modelos baseados na arquitetura *transformer* processam texto em janelas fixas, impossibilitando a análise completa de documentos extensos em uma única passada.

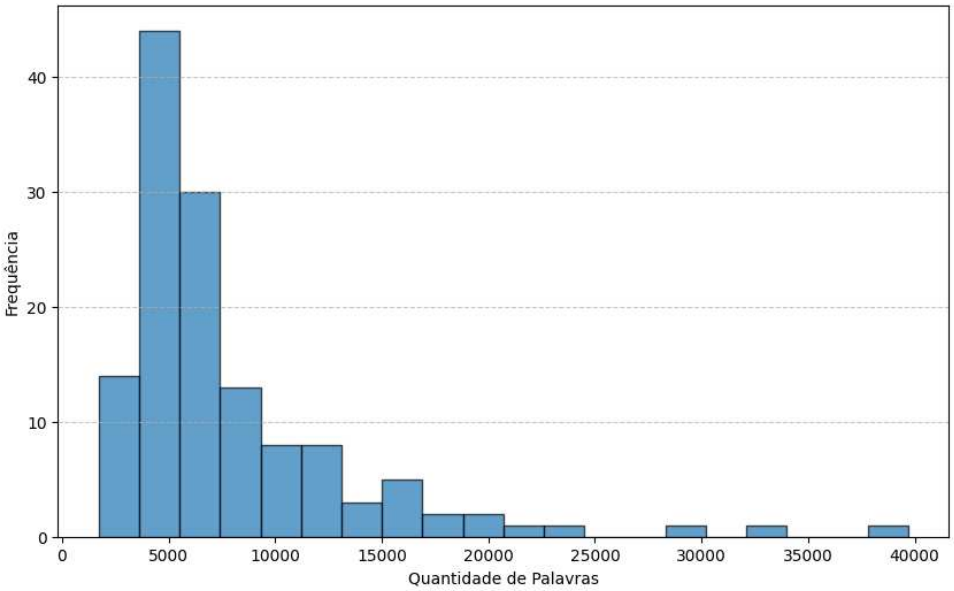

Figura 6 – Histograma da quantidade de palavras por documento
Fonte: Elabordado pelo autor (2024)

Os documentos analisados apresentam um tamanho médio de 7939,31 palavras, significativamente superior à janela de contexto suportada pelos modelos. A Figura 6 apresenta o histograma da distribuição do número de palavras por documento, evidenciando a variação no comprimento dos textos analisados. Para lidar com essa limitação, adotou-se o truncamento do texto, priorizando as seções iniciais, onde geralmente estão informações-chave, como a qualificação das partes e a descrição do objeto da ação. Não foram aplicadas técnicas de segmentação do texto em múltiplas janelas nem resumo automático antes da vetorização.

4.1.5 Modelagem

O *framework* proposto tem como objetivo otimizar a recomendação de possíveis casos de litigância predatória em processos protocolados no PJe, garantindo uma classificação mais precisa. Essa abordagem busca reduzir ineficiências na gestão processual, minimizar prejuízos éticos e diminuir os custos operacionais do setor jurídico. Dessa forma,

este estudo contribui para um melhor aproveitamento dos recursos dos tribunais, ao explorar soluções modernas e eficientes para o aprimoramento do Judiciário.

Nesta etapa, foi desenvolvido um *framework* experimental para a recuperação de processos semanticamente similares. Para isso, foram investigados e comparados diferentes vetorizadores de modelos de linguagem pré-treinados, com o objetivo de identificar quais geram as representações vetoriais mais adequadas para essa tarefa. Além disso, foi criado um ambiente de avaliação para validar o desempenho dos vetorizadores utilizados no *framework*, conforme detalhado na Seção 4.1.6.

A Figura 7 ilustra o fluxo de atividades *framework*, dividido em quatro etapas principais: (1) Preparação da Base de Dados; (2) Cálculo de Similaridade; e (3) Saída.

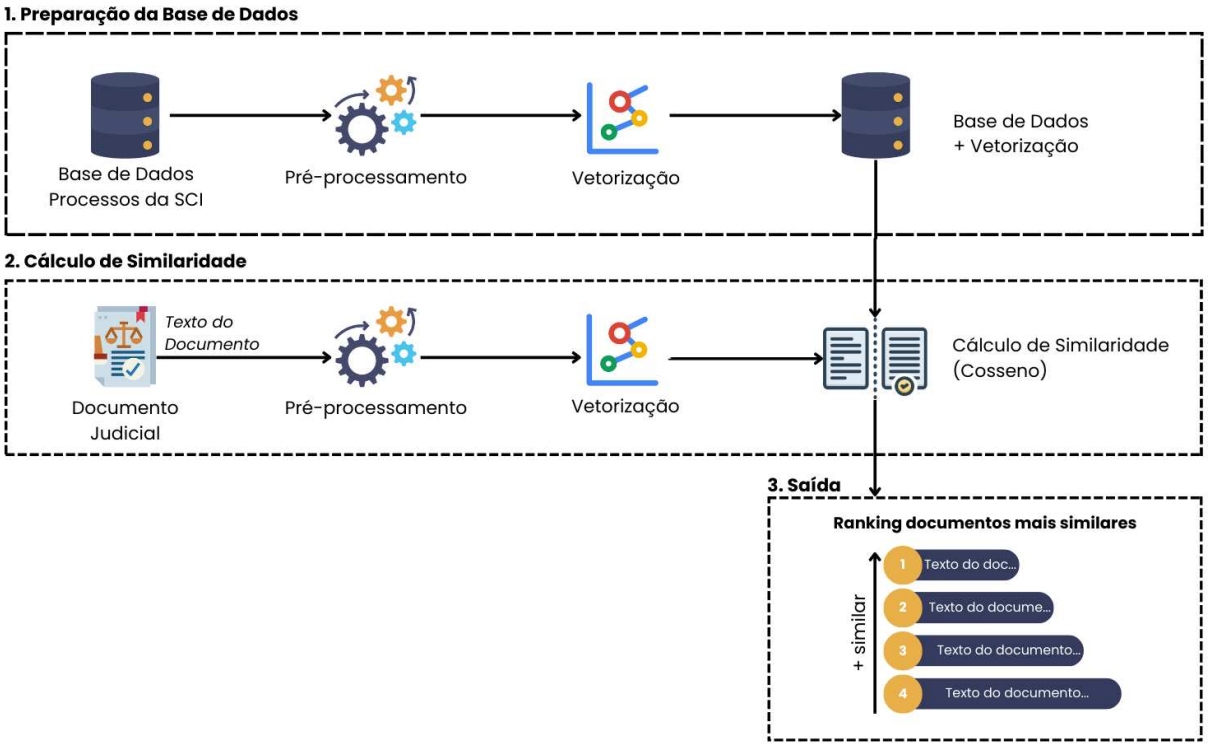

Figura 7 – *framework* proposto para recomendação de processos de litigância predatória utilizando similaridade semântica
Fonte: Elaborado pelo autor (2024)

Inicialmente, realiza-se a preparação dos dados, etapa que envolve o pré-processamento, a vetorização e o armazenamento do conjunto de dados. O pré-processamento aplica as técnicas descritas na Seção 4.1.4. Em seguida, a vetorização constrói a base de consulta a partir dos vetorizadores.

Para este trabalho, foram utilizados modelos que incluem versões generalistas do *BERT*, treinadas com um *corpus* em português, como o *RoBERTa ptBR* e o *BERTimbau*. Além disso, também foram empregadas versões do *BERT* adaptadas especificamente para o contexto jurídico, como o *BERTikal* e o *BumbaBERT Small*.

Neste trabalho, foram investigados modelos de linguagem contextuais por sua adequação à realização de buscas semânticas, alinhando-se aos objetivos propostos. Dessa forma, busca-se gerar *embeddings* contextuais para aplicação na tarefa de RI. Outros métodos, como o *Term Frequency - Inverse Document Frequency* (TF-IDF), poderiam ser utilizados para a geração de representações vetoriais. No entanto, por se basear na técnica *bag-of-words*, o TF-IDF desconsidera a ordem e o contexto das palavras, capturando apenas a frequência de termos nos documentos. Essa limitação torna o modelo menos adequado para tarefas que dependem da compreensão do significado das palavras em diferentes contextos (MANNING; RAGHAVAN; SCHüTZE, 2008).

O modelo *RoBERTa* foi treinado especificamente para o idioma português brasileiro (*pt-BR*) (LIU, 2019). Desenvolvido originalmente pela *Facebook AI*, seu objetivo foi aprimorar o desempenho do *BERT*. Sua arquitetura é baseada no modelo *Transformer*, utilizando camadas de atenção bidirecionais para capturar nuances do texto. Além disso, foi treinado com um *corpus* extenso da língua portuguesa, incluindo fontes de notícias, redes sociais, artigos acadêmicos e legislações. A escolha do *RoBERTa* neste trabalho se deve aos resultados alcançados na pesquisa de (OLIVEIRA; NASCIMENTO, 2022).

O *BERTimbau* (SOUZA; NOGUEIRA; LOTUFO, 2020) segue a arquitetura padrão do *BERT* em sua versão *base*. Esse modelo foi pré-treinado com um grande conjunto de textos em português disponíveis publicamente. Como foi desenvolvido especificamente para o idioma português, seus parâmetros e pesos foram otimizados, proporcionando maior eficiência em tarefas desse idioma em comparação com o *BERT* original.

Entre os modelos adaptados para domínios específicos, destaca-se o *BERTikal* (POLO et al., 2021), treinado especificamente para o domínio jurídico brasileiro. Seu treinamento foi realizado a partir de um *checkpoint* do *BERTimbau*, utilizando documentos jurídicos brasileiros, resultando em um modelo especializado para o contexto legal.

Por fim, o *BumbaBERT Small* (CARMO, 2024) foi desenvolvido no âmbito de um Acordo de Cooperação Técnica voltado para tarefas do judiciário brasileiro. Seu treinamento utilizou um *corpus* de petições iniciais do TJMA. A versão *Small* foi escolhida por sua eficiência computacional, garantindo uma execução mais rápida sem comprometer o desempenho. Além disso, esse modelo demonstrou alto desempenho em tarefas de PLN, como a classificação de documentos jurídicos. Os dados vetorizados foram armazenados em um *dataframe*, visando facilitar a implementação das consultas.

Na implementação dos modelos baseados em BERT, não foi realizada uma análise específica quanto à diferença entre versões *cased* e *uncased*. Durante o pré-processamento dos dados, todas as palavras foram convertidas para minúsculas, o que efetivamente tornou a capitalização irrelevante para o treinamento e avaliação dos modelos. Embora essa abordagem possa reduzir a variabilidade do vocabulário e melhorar o desempenho em alguns contextos, ela também pode impactar a identificação de termos jurídicos específicos

que fazem uso de capitalização, como siglas e nomes próprios.

A partir dos dados vetorizados, inicia-se a etapa de cálculo de similaridade. O processo tem início com a entrada de um documento, que passa pelos mesmos procedimentos aplicados na base de consulta na primeira etapa do *framework*. Em seguida, calcula-se a similaridade entre o documento de entrada e os demais documentos armazenados, utilizando a métrica de similaridade do cosseno. Com isso, gera-se um *ranking* dos documentos mais relevantes, ordenados de acordo com seu grau de similaridade semântica em relação ao documento consultado. Na etapa final, o *framework* gera a recomendação de processos que possivelmente configuram litigância predatória, com base na similaridade semântica dos documentos analisados.

4.1.6 Avaliação

Nesta seção, descrevemos o processo de avaliação adotado para comparar os diferentes modelos de *embeddings* utilizados no *framework* de recuperação de documentos semanticamente similares. O objetivo da avaliação foi medir a eficácia de cada modelo na tarefa de recuperar corretamente os documentos relevantes dentro de um *ranking* baseado na similaridade do cosseno.

A avaliação foi realizada utilizando um conjunto de 13 documentos de consulta, onde cada documento possuía um único documento relevante na base de consulta previamente identificado por especialistas. A base de consulta foi composta por 107 documentos vetorizados fixos. Para cada documento de consulta, o *framework* gerou um *ranking* de documentos ordenados pela similaridade do cosseno.

Foram comparados quatro modelos de *embeddings*, e para cada modelo, realizamos 13 consultas. As métricas utilizadas para avaliar os modelos estão descritas na subseção 2.5.2.2. Apesar de diferentes métricas terem sido calculadas, neste estudo o desempenho dos modelos foi medido utilizando as métricas *Recall@k* e *MAP*, por serem as mais adequadas para o contexto do *framework* desenvolvido (COSTA, 2024).

O *Recall@k* foi escolhido porque mede a proporção de itens relevantes recuperados entre os *top* k resultados, o que é essencial para garantir que o modelo consiga recuperar a maior quantidade possível de itens relevantes. No entanto, também é importante considerar a precisão dos resultados, pois, à medida que k aumenta, cresce a quantidade de ruído nos resultados, ou seja, mais itens irrelevantes podem ser incluídos. Por isso, a métrica *MAP* também será avaliada, pois sintetiza a relação entre precisão e revocação em um único valor, permitindo uma avaliação mais equilibrada da qualidade do ranqueamento gerado pelos modelos.

4.1.7 Definição do Valor de *k*

Para o cálculo das métricas de avaliação, é necessário definir o valor de k, que representa a quantidade de elementos considerados nos *rankings* gerados. Contudo, não há um consenso na literatura sobre sua definição, pois esse valor depende do domínio da aplicação e do tamanho da base de documentos (BAEZA-YATES; RIBEIRO-NETO, 2013). Diferentes estudos propõem valores variados para k, a depender do contexto, o trabalho de Costa (2024) recomenda k=100 para recuperação de documentos similares, enquanto Manning, Raghavan e Schütze (2008) explora valores menores, como 10 e 20. No entanto, esses valores podem não ser adequados para todos os cenários, especialmente quando a base de documentos é reduzida, como neste estudo.

Por outro lado, Salton e Buckley (1988) defendem a avaliação de múltiplos valores de k, permitindo que diferentes cenários de aplicação sejam testados. Seguindo essa abordagem, foram realizados testes experimentais variando o valor de k de 5 até 25, em intervalos de 5 unidades. Para cada valor de k, foram calculados os valores médios da métrica *Recall@k* ao longo das 13 consultas, permitindo observar a evolução do desempenho do modelo conforme o número de documentos recuperados aumentava.

Esse processo possibilitou a identificação de um ponto adequado para a avaliação do *ranking* gerado. Os detalhes sobre a escolha final do valor de k e os resultados obtidos serão apresentados na Seção 5.1.

4.1.8 Geração dos *Rankings*

O processo de recuperação foi conduzido para um conjunto de 13 consultas, seguindo uma sequência de etapas estruturadas. Inicialmente, cada documento de consulta foi transformado em um vetor de *embedding*, utilizando o modelo em avaliação. Após a conversão para *embeddings*, foi calculada a similaridade do cosseno entre o vetor da consulta e os vetores correspondentes aos 107 documentos da base. Com os valores de similaridade obtidos, os documentos foram ordenados em ordem decrescente, formando um *ranking* de relevância. Para a avaliação da recuperação, foram considerados apenas os *top-k* primeiros documentos do *ranking*, conforme o valor de k previamente definido. Por fim, esse processo foi repetido para cada um dos 13 documentos de consulta, gerando 13 *rankings* distintos para cada modelo avaliado. Dessa forma, foi possível comparar o desempenho dos modelos testados, analisando sua capacidade de recuperar documentos relevantes a partir das consultas fornecidas.

4.1.9 Cálculo das Métricas

Para cada modelo, as métricas *Recall@k* e *MAP* foram calculadas para as 13 consultas e, posteriormente, foi determinada a média e o desvio padrão dessas métricas.

4.1.9.1 Implementação do Recall@k

A métrica *Recall@k* foi utilizada para verificar se o documento relevante apareceu nos *top-k* primeiros resultados do *ranking*. O cálculo seguiu a seguinte definição:

$$Recall@k = \begin{cases} 1, & \text{se o documento relevante está entre os k primeiros resultados} \\ 0, & \text{caso contrário} \end{cases} \tag{4.1}$$

Ao final, o *Recall@k* médio foi obtido calculando a média dos valores de *Recall@k* em todas as 13 consultas. Além disso, foi calculado o desvio padrão para avaliar a variação dos resultados entre as consultas.

4.1.9.2 Implementação do MAP

A métrica *Mean Average Precision (MAP)* mede a precisão média ao longo das posições onde documentos relevantes aparecem no *ranking*. No entanto, como cada consulta com a base de teste possui apenas um documento relevante, o *MAP* se reduz ao cálculo da precisão na posição em que esse documento aparece.

Conforme descrito na subseção 2.5.2.2 o *MAP* (Equação. 2.10) é definido como a média da *AP* para todas as consultas, 13 neste caso. Como no presente estudo cada consulta tem apenas um documento relevante, a fórmula do *AP* se simplifica para Equação 4.2.

$$AP = P(rank_{relevante}) \tag{4.2}$$

Ou seja, o *AP* é simplesmente a precisão no ponto em que o documento relevante aparece no *ranking*. Ao final, o *MAP* é obtido calculando a média dos valores de *AP* em todas as 13 consultas.

4.1.10 Configuração da Implementação

O código-fonte utilizado para geração das representações vetoriais e execução dos cálculos de similaridade foi implementado utilizando um ambiente virtual (*venv*) linguagem de programação *Python* na versão 3.12, nos quais foram utilizadas bibliotecas como *Pandas*, *scikit-learn*, *Matplotlib*, *NumPy* e *torchmetrics*. O código foi executado em um computador pessoal em uma CPU com 6 núcleos de 4,4GHz e 24GB de RAM.

As representações vetoriais foram geradas utilizando os parâmetros descritos na Tabela 9.

Tabela 9 – Parâmetros utilizados para a geração das representações vetoriais.

Parâmetro	**Descrição**
return_tensors	Definido como verdadeiro para retornar os tensores no formato do *PyTorch*.
truncation	Serve para cortar o texto se for maior que o tamanho máximo (512 *tokens* no *BERT*).
padding	Adiciona *padding* para igualar o comprimento das sequências.
max_length	Define o número máximo de *tokens* processados.

4.1.11 Entrega

O produto final deste trabalho consiste na implementação e validação de um *framework* experimental voltado à recomendação de possíveis casos de litigância predatória por meio da análise de similaridade semântica entre documentos jurídicos. A entrega inclui a descrição detalhada da abordagem adotada, os modelos utilizados, os experimentos realizados e a análise quantitativa dos resultados.

Para avaliar a eficácia do *framework*, serão consideradas métricas de desempenho como *Recall@k* e *MAP*. Além disso, serão discutidas as limitações da abordagem e possíveis melhorias. Os resultados obtidos contribuirão para o aprimoramento de mecanismos de recuperação de informação no campo jurídico, fornecendo subsídios para o desenvolvimento de ferramentas que auxiliem na detecção de padrões recorrentes associados à litigância predatória.

5 RESULTADOS E DISCUSSÃO

Neste capítulo, são apresentados os resultados obtidos a partir da avaliação dos modelos de *embeddings* no contexto da recuperação de documentos semanticamente similares. A análise está dividida em duas partes principais: (1) a definição do valor de k mais adequado para as métricas de avaliação e (2) a comparação da eficiência dos modelos testados. As métricas utilizadas para a avaliação foram *Recall@k* e MAP, conforme descrito na Seção 2.5.2.2.

5.1 Escolha do valor *k*

A escolha do valor de k é fundamental para a avaliação do desempenho do sistema de recuperação de documentos, pois influencia diretamente as métricas baseadas em corte como *Recall@k* e *Precision@k*, e indiretamente outras, como *MAP* e *MRR*. A Figura 8 mostra a evolução das métricas *Recall@k* e *MAP* em função de k para cada modelo testado.

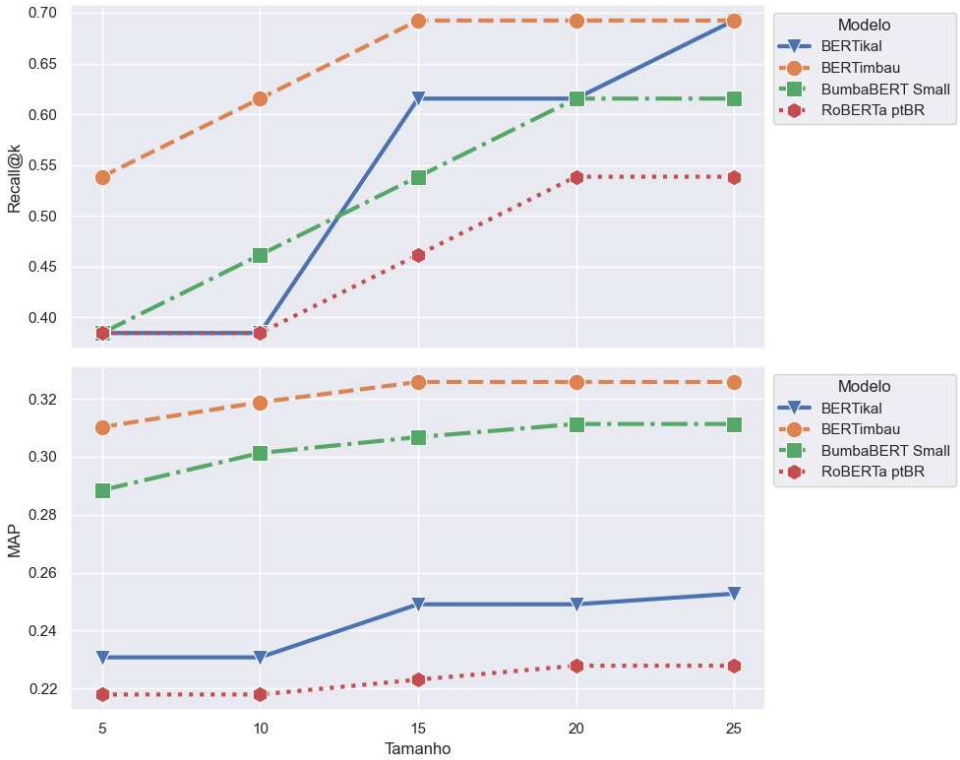

Figura 8 – Evolução do *Recall@k* e *MAP* em função de k.

A análise da Figura 8 evidencia um comportamento esperado para o *Recall@k*, que cresce com o aumento de k, uma vez que um número maior de documentos é recuperado. Observa-se que a métrica tende a se estabilizar a partir de k=20 para a maioria dos modelos.

Essa estabilização sugere que, ao atingir esse valor, a inclusão de mais documentos nos *rankings* não contribui significativamente para a recuperação de itens relevantes. No entanto, verifica-se também que o modelo *BERTikal* continua apresentando um leve crescimento até $k = 25$. Esse comportamento pode indicar que o *BERTikal* está recuperando documentos relevantes de forma mais dispersa, o que pode comprometer a priorização dos documentos no *ranking*.

Por outro lado, a métrica *MAP*, apresenta um crescimento mais sutil à medida que k aumenta, mostrando que a inclusão de mais documentos no *ranking* não impacta tanto a qualidade da ordenação dos dos documentos retornados. Com base nesses resultados, foi definido $k = 20$ para as discussões subsequentes na Seção 5.2, pois esse valor oferece um equilíbrio entre alta recuperação de itens relevantes e boa qualidade de ranqueamento, sem incluir muitos documentos irrelevantes.

5.2 Eficácia dos Modelos

Na Tabela 10, são apresentados os resultados comparativos dos modelos de *embeddings* em relação às métricas *Recall@k* e *MAP*, considerando o k escolhido.

Tabela 10 – Desempenho dos modelos de linguagem baseados no BERT

Modelo (k=5)	***Precision@k*** (±dp)	***Recall@k*** (±dp)	***MAP*** (***k***) (±dp)	***MRR*** (***k***) (±dp)
BERTikal	0,0769 ± 0,0253	0,3846 ± 0,5064	0,2308 ± 0,3637	0,2308 ± 0,3637
BERTimbau	**0,1077 ± 0,0240**	**0,5385 ± 0,4804**	**0,3103 ± 0,3585**	**0,3103 ± 0,3585**
BumbaBERT Small	0,0769 ± 0,0253	0,3846 ± 0,5064	0,2885 ± 0,4172	0,2885 ± 0,4172
RoBERTa ptBR	0,0769 ± 0,0259	0,3846 ± 0,5189	0,2179 ± 0,3155	0,2179 ± 0,3155
Modelo (k=10)				
BERTikal	0,0385 ± 0,0253	0,3846 ± 0,5064	0,2308 ± 0,3637	0,2308 ± 0,3637
BERTimbau	**0,0615 ± 0,0240**	**0,5385 ± 0,4804**	**0,3188 ± 0,3585**	**0,3188 ± 0,3585**
BumbaBERT Small	0,0462 ± 0,0253	0,4615 ± 0,5064	0,3068 ± 0,4172	0,3068 ± 0,4172
RoBERTa ptBR	0,0385 ± 0,0259	0,3846 ± 0,5189	0,2179 ± 0,3155	0,2179 ± 0,3155
Modelo (k=15)				
BERTikal	0,0410 ± 0,0253	0,6154 ± 0,5064	0,2491 ± 0,3637	0,2491 ± 0,3637
BERTimbau	**0,0462 ± 0,0240**	**0,6923 ± 0,4804**	**0,3258 ± 0,3585**	**0,3258 ± 0,3585**
BumbaBERT Small	0,0359 ± 0,0253	0,5385 ± 0,5064	0,3068 ± 0,4172	0,3068 ± 0,4172
RoBERTa ptBR	0,0308 ± 0,0259	0,4615 ± 0,5189	0,2231 ± 0,3155	0,2231 ± 0,3155
Modelo (k=20)				
BERTikal	0,0308 ± 0,0253	0,6154 ± 0,5064	0,2491 ± 0,3637	0,2491 ± 0,3637
BERTimbau	**0,0346 ± 0,0240**	**0,6923 ± 0,4804**	**0,3258 ± 0,3585**	**0,3258 ± 0,3585**
BumbaBERT Small	0,0308 ± 0,0253	0,6154 ± 0,5064	0,3113 ± 0,4172	0,3113 ± 0,4172
RoBERTa ptBR	0,0269 ± 0,0259	0,5385 ± 0,5189	0,2279 ± 0,3155	0,2279 ± 0,3155
Modelo (k=25)				
BERTikal	0,0277 ± 0,0253	0,6923 ± 0,5064	0,2527 ± 0,3637	0,2527 ± 0,3637
BERTimbau	**0,0277 ± 0,0240**	**0,6923 ± 0,4804**	**0,3258 ± 0,3585**	**0,3258 ± 0,3585**
BumbaBERT Small	0,0246 ± 0,0253	0,6154 ± 0,5064	0,3113 ± 0,4172	0,3113 ± 0,4172
RoBERTa ptBR	0,0215 ± 0,0259	0,5385 ± 0,5189	0,2279 ± 0,3155	0,2279 ± 0,3155

Fonte: Elaborado pelo Autor (2025).

Como pode ser observado na Tabela 10, o modelo *BERTimbau* obteve o melhor desempenho em ambas as métricas, com um *Recall@k* de 0,6923 e um *MAP* de 0,3258.

Esse resultado sugere que o *BERTimbau* não apenas recupera mais documentos relevantes, mas também os posiciona entre os primeiros do *ranking*.

O segundo melhor desempenho foi obtido pelo *BumbaBERT Small*, que atingiu um *Recall@k* de 0,6154 e um *MAP* de 0,3113. Embora tenha um desempenho ligeiramente inferior ao *BERTimbau*, esse modelo demonstrou ser competitivo e pode ser uma alternativa eficiente, especialmente em cenários com restrições computacionais, devido ao seu tamanho reduzido (CARMO, 2024). O modelo *BumbaBERT Small* apresentou um desempenho próximo, mas ligeiramente inferior, com um *Recall@k* de 0,6154 e um *MAP* de 0,3113. Por outro lado, os modelos *BERTikal* e *RoBERTa ptBR* tiveram desempenhos inferiores, com o *RoBERTa ptBR* sendo o menos eficaz, apresentando um *Recall@k* de apenas 0,5385 e um *MAP* de 0,2279.

A vantagem do *BERTimbau* pode ser atribuída a diferentes fatores. Primeiramente, trata-se de um modelo treinado especificamente para a língua portuguesa, o que pode conferir uma melhor adaptação ao processamento textual em comparação com modelos que passaram por adaptação ou ajustes finos para domínios específicos. No entanto, essa justificativa isolada não explica completamente os resultados, visto que o *RoBERTa ptBR*, também ajustado para o português, apresentou o pior desempenho.

Outro ponto a considerar é a natureza dos dados utilizados para o treinamento dos modelos. O *BERTikal*, apesar de ter sido treinado com dados jurídicos do Brasil, pode ter uma forma de representação textual menos eficiente para a tarefa específica de recuperação de informação semântica, o que pode ter impactado seu desempenho. Isso sugere que a simples especialização no domínio jurídico não garante automaticamente um desempenho superior, especialmente quando o modelo é aplicado a um conjunto de documentos traduzidos. Como todos os modelos foram testados com os mesmos documentos traduzidos, a hipótese de que a tradução afetou apenas o *BERTikal* não se sustenta. Caso fosse um fator determinante, modelos como o *BumbaBERT Small,* que também tem treinamento jurídico, deveriam ter sofrido impacto similar, mas seu desempenho foi superior ao do *BERTikal*.

Outro fator relevante é que o *RoBERTa ptBR* foi treinado com um volume maior de dados e sem a etapa de NSP, uma característica que visa aprimorar a representação contextual dos textos. No entanto, a eliminação do NSP não é exclusiva desse modelo e, por si só, pode não explicar seu desempenho inferior. Além disso, o *RoBERTa ptBR* foi treinado em um *corpus* amplo e generalista, sem uma especialização direta em textos jurídicos.

Contudo, o fato do *BERTimbau*, que também foi treinado em um *corpus* amplo e não especializado no domínio jurídico, ter apresentado o melhor desempenho indica que a especialização jurídica não é necessariamente um fator determinante para o sucesso na tarefa. Isso sugere que a arquitetura e a forma como cada modelo captura a semântica dos

textos podem ter um papel mais significativo do que a especialização no domínio jurídico. Assim, os resultados reforçam que o alinhamento entre o pré-treinamento do modelo e a tarefa específica é importante, mas não é o único fator que influencia o desempenho. Aspectos como a estratégia de treinamento, a qualidade e diversidade do *corpus*, e a maneira como o modelo aprende a representação semântica devem ser considerados para entender melhor as diferenças de desempenho entre os modelos.

A análise do desvio padrão dos resultados revela que o *RoBERTa ptBR* apresentou a maior variabilidade na métrica *Recall@20* (±51,89%), enquanto o *BERTimbau* teve a menor variação (±48,04%), indicando que este último possui um desempenho mais estável na recuperação de documentos jurídicos. Além disso, em relação à métrica *MAP*, o *RoBERTa ptBR* foi o modelo com menor variação (±31,55%), enquanto o *BumbaBERT Small* apresentou o maior desvio padrão (±41,74%), sugerindo que este modelo pode ser mais sensível à estrutura dos documentos consultados, favorecendo algumas buscas específicas, mas sendo menos consistente em outras.

6 CONSIDERAÇÕES FINAIS

Neste capítulo, são apresentadas as considerações finais do estudo, destacando as contribuições alcançadas, as limitações identificadas e as perspectivas para trabalhos futuros.

Este trabalho investigou a viabilidade e implementação de um *framework* baseado em RI para apoiar a análise de demandas predatórias no sistema judiciário. A abordagem proposta, fundamentada no uso de *embeddings* contextuais e técnicas de similaridade semântica, demonstrou potencial para otimizar a busca de documentos relevantes e contribuir para a detecção de padrões característicos da litigância predatória. O estudo foi guiado pela metodologia *CRISP-DM*, permitindo um desenvolvimento estruturado e iterativo, no qual foi projetada, implementada e avaliada uma solução para aprimorar a identificação de demandas abusivas no Judiciário brasileiro, especificamente em processos relacionados a empréstimos consignados.

Os experimentos realizados evidenciaram que o modelo *BERTimbau* apresentou o melhor desempenho na recuperação de documentos semanticamente similares, alcançando um *Recall@k* de 0,6923 e um *MAP* de 0,3258. Esses resultados indicam que esse modelo pode ser uma alternativa viável para a análise de similaridade textual no domínio jurídico, especialmente quando comparado a outros modelos testados, como *BERTikal*, *RoBERTa ptBR* e *BumbaBERT Small*.

Um aspecto relevante observado nos resultados foi a equivalência entre os valores das métricas *MAP* e *MRR* para todos os modelos avaliados. Esse comportamento se deve à característica do *ground truth* utilizado, onde cada consulta possuía apenas um único documento relevante. Como o *MAP* mede a precisão média das posições dos documentos relevantes no *ranking*, e o *MRR* considera apenas a posição do primeiro documento relevante, os valores dessas métricas permaneceram idênticos. Para avaliações futuras, sugere-se o uso de uma base de dados com múltiplos documentos relevantes por consulta, permitindo uma análise mais abrangente da qualidade do ranqueamento gerado pelos modelos.

Além da avaliação experimental, o presente estudo também buscou contribuir para pesquisas em PLN aplicadas ao Judiciário, explorando soluções inovadoras para os desafios específicos do setor. A abordagem proposta poderá ser utilizada como um complemento ao robô Nirie do TJMA, auxiliando na melhoria das predições relacionadas à litigiosidade predatória nos processos judiciais.

6.1 Limitações

O desenvolvimento deste trabalho enfrentou algumas limitações, principalmente relacionadas à disponibilidade de dados no domínio jurídico brasileiro. Embora tenha sido obtida uma base de processos do TJMA, ela não possuía um *ground truth* estruturado para a avaliação quantitativa do *framework* proposto. Assim, foi necessário recorrer a uma base de dados externa, proveniente do sistema judiciário da Índia. Essa limitação compromete a generalização dos resultados para o contexto do Judiciário brasileiro, uma vez que as particularidades linguísticas e estruturais dos documentos podem diferir significativamente.

Diferentemente do estudo de Petrović e Stanković (2019), este trabalho não realizou uma análise comparativa do impacto do pré-processamento na qualidade dos resultados. Essa ausência de experimentação específica configura uma limitação do estudo, uma vez que não se pode afirmar com certeza se as técnicas adotadas tiveram um efeito positivo ou negativo na tarefa de similaridade semântica no contexto jurídico.

Outra limitação importante é a restrição de 512 *tokens* imposta pelos modelos baseados no *BERT*, o que pode impactar a análise de textos jurídicos extensos. Em muitos casos, documentos processuais possuem um grande volume de informações relevantes que podem ser descartadas devido a essa limitação. Modelos mais avançados, como aqueles que suportam 1024 *tokens*, poderiam mitigar essa questão, mas a um custo computacional mais elevado.

Outra limitação deste estudo é o possível viés introduzido pela tradução automática de alguns documentos para o português. Como os modelos avaliados foram treinados nessa língua, diferenças na estrutura textual e eventuais perdas semânticas podem ter influenciado a representação dos textos e o desempenho dos modelos. Assim, a comparação entre eles deve ser interpretada com cautela.

Além disso, a orientação do estudo pelo *CRISP-DM* permitiu que as decisões metodológicas fossem adaptadas com base em evidências experimentais. No entanto, a ausência de testes diretos com processos reais do Judiciário brasileiro impediu uma avaliação mais concreta da aplicabilidade do *framework* em um cenário prático.

6.2 Trabalhos Futuros

Conforme mencionado na Seção 4.1.2, a base de dados do TJMA não pôde ser utilizada diretamente na avaliação dos modelos devido à ausência de um *ground truth* adequado para a tarefa de recuperação de documentos. Como trabalho futuro, sugere-se a implementação de um *pipeline* de curadoria e anotação dessa base, no qual especialistas jurídicos categorizariam manualmente um subconjunto de processos como litigância predatória ou não, além de identificar documentos similares. Esse processo

poderia ser conduzido em parceria com o próprio tribunal, envolvendo juízes, servidores e pesquisadores da área jurídica. Com um conjunto de dados anotado, seria possível validar os modelos de recuperação de informação e explorar abordagens supervisionadas para a identificação de padrões na litigância predatória.

A estratégia de vetorização também pode ser aprimorada para lidar com a limitação de 512 *tokens* dos modelos baseados no *BERT*. A técnica de *chunking*, que divide o documento em segmentos menores antes da vetorização, poderia permitir representar de forma mais completa o conteúdo dos processos, melhorando a recuperação de informações relevantes. Além disso, novos modelos de PLN, como aqueles baseados em *Transformers* mais avançados, como *Longformer* e *BigBird*, podem ser explorados para lidar melhor com documentos longos. Essas arquiteturas foram projetadas para processar grandes quantidades de texto sem perder contexto e poderiam aprimorar a análise de litigância predatória.

Os modelos de linguagem utilizados neste estudo geram representações vetoriais de alta dimensionalidade, com *embeddings* variando entre 768 e 1024 dimensões, dependendo do modelo utilizado. Embora essa alta dimensionalidade permita capturar nuances semânticas complexas, ela também pode introduzir desafios computacionais, como aumento no tempo de processamento e necessidade de maior capacidade de armazenamento. Em pesquisas futuras, sugere-se a investigação de técnicas de redução de dimensionalidade, como PCA (*Principal Component Analysis*) ou *t-SNE*, para avaliar se uma representação mais compacta poderia manter o desempenho dos modelos enquanto reduz custos computacionais.

Este trabalho utilizou modelos baseados em *Transformers* para representar e comparar a similaridade semântica entre processos judiciais. No entanto, uma alternativa menos custosa computacionalmente seria o uso de representações baseadas em TF-IDF, que consideram todo o documento sem a limitação de contexto imposta pelos modelos BERT. Embora TF-IDF não capture o significado semântico das palavras no mesmo nível dos *embeddings* contextuais, ele pode ser útil para a detecção de padrões estruturais e terminológicos nos textos jurídicos. Como trabalho futuro, sugere-se uma análise comparativa entre esses métodos, considerando as vantagens e limitações de cada abordagem para a recuperação de informações jurídicas e a identificação de litigância predatória.

Outro avanço possível é a integração do *framework* proposto a sistemas já utilizados no Judiciário, como o robô Nirie do TJMA. Essa adaptação permitiria avaliar sua aplicabilidade prática, automatizando a triagem de processos de maneira mais eficiente e precisa. Dessa forma, este trabalho contribui para o avanço das pesquisas em PLN no contexto jurídico, propondo uma abordagem inovadora para a detecção de padrões de litigância predatória. Embora ainda existam desafios a serem superados, os resultados obtidos indicam que técnicas de Recuperação de Informação baseada em similaridade

semântica podem desempenhar um papel significativo na modernização do Judiciário e na redução de demandas abusivas.

A comparação entre os modelos foi realizada com base em métricas de desempenho numéricas, sem a aplicação de testes estatísticos para avaliar a significância das diferenças observadas. Como os resultados foram relativamente próximos, não é possível afirmar com rigor estatístico qual modelo é de fato superior para a tarefa proposta. Para trabalhos futuros, sugere-se a realização de testes de significância, como *t-test* ou *Wilcoxon signed-rank test*, para validar se as diferenças entre os modelos são estatisticamente relevantes, proporcionando uma avaliação mais robusta na escolha do melhor modelo.

Referências

ARAÚJO, V. S. de; GABRIEL, A. de P.; PORTO, F. R. Justiça 4.0: a transformação tecnológica do poder judiciário deflagrada pelo cnj no biênio 2020-2022. *Revista Eletrônica Direito Exponencial-DIEX*, v. 1, n. 1, p. 1–18, 2022. Citado na página 13.

ASPERTI, M. C. de A.; SILVA, P. E. A. da; GABBAY, D. M.; COSTA, S. H. da. Why the "haves" come out ahead in brazil? revisiting speculations concerning repeat players and one-shooters in the brazilian litigation setting. *Direito Público*, v. 16, n. 88, 2019. Citado na página 12.

BAEZA-YATES, R.; RIBEIRO-NETO, B. *Recuperação de Informação-: Conceitos e Tecnologia das Máquinas de Busca*. [S.l.]: Bookman Editora, 2013. Citado na página 48.

BARRETO, N. M. M. *Indústria 4.0: O Impacto do Big Data e Internet of Things*. Dissertação (Mestrado) — Universidade de Lisboa (Portugal), 2019. Citado na página 12.

BENGIO, Y.; DUCHARME, R.; VINCENT, P. A neural probabilistic language model. *Advances in neural information processing systems*, v. 13, 2000. Citado na página 24.

BHATTACHARYA, P.; GHOSH, K.; PAL, A.; GHOSH, S. Legal case document similarity: You need both network and text. *Information Processing & Management*, Elsevier, v. 59, n. 6, p. 103069, 2022. Citado 6 vezes nas páginas 8, 23, 34, 36, 41 e 42.

BISHOP, C. M.; NASRABADI, N. M. *Pattern recognition and machine learning*. [S.l.]: Springer, 2006. v. 4. Citado na página 21.

BRAGANÇA, F.; BRAGANÇA, L. F. da F. Revolução 4.0 no poder judiciário: levantamento do uso de inteligência artificial nos tribunais brasileiros. *Revista da Seção Judiciária do Rio de Janeiro*, v. 23, n. 46, p. 65–76, 2019. Citado na página 13.

BROWN, T.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J. D.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A. et al. Language models are few-shot learners. *Advances in neural information processing systems*, v. 33, p. 1877–1901, 2020. Citado na página 22.

BUCKLEY, C.; VOORHEES, E. M. Evaluating evaluation measure stability. In: ACM NEW YORK, NY, USA. *ACM SIGIR Forum*. [S.l.], 2017. v. 51, n. 2, p. 235–242. Citado na página 31.

CARMO, F. A. do. *Representações Embeddings Orientadas à Linguagem Jurídica Brasileira*. 84 f. Dissertação (Mestrado em Engenharia da Compuatação e Sistemas) — Universidade Estadual do Maranhão, São Luís - MA, 2024. Citado 6 vezes nas páginas 13, 32, 34, 36, 46 e 53.

CASELI, H. d. M.; NUNES, M. d. G. V. Processamento de linguagem natural: conceitos, técnicas e aplicações em português. 2024. Citado 3 vezes nas páginas 25, 26 e 27.

CHOWDHARY KR1442, K. Natural language processing. *Fundamentals of artificial intelligence*, Springer, p. 603–649, 2020. Citado na página 20.

CIRQUEIRA, D.; PINHEIRO, M. F.; JACOB, A.; LOBATO, F.; SANTANA, Á. A literature review in preprocessing for sentiment analysis for brazilian portuguese social media. In: IEEE. *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. [S.l.], 2018. p. 746–749. Citado 2 vezes nas páginas 21 e 43.

CNJ. *Solução tecnológica auxilia no combate à litigância predatória na Paraíba*. 2025. Disponível em: <https://www.cnj.jus.br/solucao-tecnologica-auxilia-no-combate-a-litigancia-predatoria-na-paraiba/>. Citado na página 14.

COSTA, W. M. Similaridade semântica entre acórdãos para apoio na formulação de jurisprudência do tcu. 2024. Citado 6 vezes nas páginas 17, 33, 36, 40, 47 e 48.

DAVE, K.; LAWRENCE, S.; PENNOCK, D. *Proceedings of the 12th international conference on World Wide Web*. [S.l.]: ACM New York, NY, 2003. Citado na página 20.

FEDERAL, C. da J. *Manual de Referência - Processo Judicial Eletrônico (PJe)*. [S.l.], 2011. Acessado em: 5 jan. 2025. Disponível em: <https://www.cjf.jus.br/observatorio/arq/manual_referencia_pje.pdf>. Citado na página 12.

FILHO, M. S. M.; JUNQUILHO, T. A. Projeto victor: perspectivas de aplicação da inteligência artificial ao direito. *Revista de Direitos e Garantias Fundamentais*, Faculdade de Direito de Vitória, v. 19, n. 3, p. 218–237, 2018. Citado 3 vezes nas páginas 12, 20 e 21.

GOLDBERG, Y. *Neural network methods in natural language processing*. [S.l.]: Morgan & Claypool Publishers, 2017. Citado na página 24.

HIRSCHBERG, J.; MANNING, C. D. Advances in natural language processing. *Science*, American Association for the Advancement of Science, v. 349, n. 6245, p. 261–266, 2015. Citado na página 20.

ILIĆ, S.; MARRESE-TAYLOR, E.; BALAZS, J. A.; MATSUO, Y. Deep contextualized word representations for detecting sarcasm and irony. *arXiv preprint arXiv:1809.09795*, 2018. Citado na página 21.

JÄRVELIN, K.; KEKÄLÄINEN, J. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, ACM New York, NY, USA, v. 20, n. 4, p. 422–446, 2002. Citado na página 30.

JOSÉ, L.; CLEMENTINO, M. B. M. Litigância predatória: Entre o acesso à justiça e os abusos sistemáticos do direito ao processo. *Cadernos de Dereito Actual*, n. 25, p. 48–74, 2024. Citado 4 vezes nas páginas 12, 13, 16 e 17.

JURAFSKY, D.; MARTIN, J. H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 2023. Citado 2 vezes nas páginas 24 e 25.

KABIR, M. S.; ALAM, M. N. Iot, big data and ai applications in the law enforcement and legal system: A review. *International Research Journal of Engineering and Technology (IRJET)*, v. 10, n. 05, p. 1777–1789, 2023. Citado na página 12.

KENTON, J. D. M.-W. C.; TOUTANOVA, L. K. Bert: Pre-training of deep bidirectional transformers for language understanding. In: MINNEAPOLIS, MINNESOTA. *Proceedings of naacL-HLT*. [S.l.], 2019. v. 1, p. 2. Citado 4 vezes nas páginas 21, 22, 24 e 25.

LAHITANI, A. R.; PERMANASARI, A. E.; SETIAWAN, N. A. Cosine similarity to determine similarity measure: Study case in online essay assessment. In: IEEE. *2016 4th International conference on cyber and IT service management*. [S.l.], 2016. p. 1–6. Citado na página 26.

LAKE, B. M.; MURPHY, G. L. Word meaning in minds and machines. *Psychological review*, American Psychological Association, v. 130, n. 2, p. 401, 2023. Citado na página 21.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015. Citado na página 20.

LIN, T.; WANG, Y.; LIU, X.; QIU, X. A survey of transformers. *AI open*, Elsevier, v. 3, p. 111–132, 2022. Citado na página 22.

LIU, Y. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, v. 364, 2019. Citado 2 vezes nas páginas 21 e 46.

MAGALHÃES, J. L.; SOUSA, I. D. V. de et al. Litigância judicial abusiva e instrumentos de gestão processual conferidos ao juiz no código de processo civil: A necessidade de preservação do direito fundamental de acesso à justiça. *REVISTA FOCO*, v. 17, n. 4, p. e4789–e4789, 2024. Citado 5 vezes nas páginas 12, 13, 16, 17 e 20.

MANNING, C. D.; RAGHAVAN, P.; SCHüTZE, H. *Introduction to Information Retrieval*. Cambridge University Press, 2008. ISBN 9780521865715. Disponível em: <https://nlp.stanford.edu/IR-book/>. Citado 5 vezes nas páginas 20, 41, 42, 46 e 48.

MARTÍNEZ-PLUMED, F.; CONTRERAS-OCHANDO, L.; FERRI, C.; HERNÁNDEZ-ORALLO, J.; KULL, M.; LACHICHE, N.; RAMÍREZ-QUINTANA, M. J.; FLACH, P. Crisp-dm twenty years later: From data mining processes to data science trajectories. *IEEE transactions on knowledge and data engineering*, IEEE, v. 33, n. 8, p. 3048–3061, 2019. Citado na página 37.

MENTZINGEN, H.; ANTÓNIO, N.; BACAO, F.; CUNHA, M. Textual similarity for legal precedents discovery: Assessing the performance of machine learning techniques in an administrative court. *International Journal of Information Management Data Insights*, Elsevier, v. 4, n. 2, p. 100247, 2024. Citado 3 vezes nas páginas 32, 34 e 36.

MOSBACH, M.; PIMENTEL, T.; RAVFOGEL, S.; KLAKOW, D.; ELAZAR, Y. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *arXiv preprint arXiv:2305.16938*, 2023. Citado na página 22.

OLIVEIRA, R. S. de; NASCIMENTO, E. G. S. Clustering by similarity of brazilian legal documents using natural language processing approaches. In: *Data Clustering*. [S.l.]: IntechOpen, 2021. p. 126. Citado na página 33.

OLIVEIRA, R. S. de; NASCIMENTO, E. G. S. Analysing similarities between legal court documents using natural language processing approaches based on transformers. *arXiv preprint arXiv:2204.07182*, 2022. Citado 6 vezes nas páginas 20, 23, 33, 34, 36 e 46.

PETROVIĆ, Đ.; STANKOVIĆ, M. The influence of text preprocessing methods and tools on calculating text similarity. *Facta Universitatis, Series: Mathematics and Informatics*, p. 973–994, 2019. Citado 4 vezes nas páginas 33, 36, 43 e 56.

POLO, F. M.; MENDONÇA, G. C. F.; PARREIRA, K. C. J.; GIANVECHIO, L.; CORDEIRO, P.; FERREIRA, J. B.; LIMA, L. M. P. de; MAIA, A. C. d. A.; VICENTE, R. Legalnlp–natural language processing methods for the brazilian legal language. *arXiv preprint arXiv:2110.15709*, 2021. Citado 4 vezes nas páginas 23, 32, 36 e 46.

RADFORD, A. Improving language understanding by generative pre-training. 2018. Citado 2 vezes nas páginas 21 e 37.

RAFFEL, C.; SHAZEER, N.; ROBERTS, A.; LEE, K.; NARANG, S.; MATENA, M.; ZHOU, Y.; LI, W.; LIU, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, v. 21, n. 140, p. 1–67, 2020. Citado 2 vezes nas páginas 21 e 24.

RAHUTOMO, F.; KITASUKA, T.; ARITSUGI, M. et al. Semantic cosine similarity. In: UNIVERSITY OF SEOUL SOUTH KOREA. *The 7th international student conference on advanced science and technology ICAST*. [S.l.], 2012. v. 4, n. 1, p. 1. Citado na página 26.

RAMPIM, T.; IGREJA, R. L. Acesso à justiça e transformação digital: Um estudo sobre o programa justiça 4.0 e seu impacto na prestação jurisdicional. *Direito Público*, v. 19, n. 102, 2022. Citado na página 13.

RODRÍGUEZ, M. M.; BEZERRA, B. L. D. Processamento de linguagem natural para reconhecimento de entidades nomeadas em textos jurídicos de atos administrativos (portarias). *Revista de Engenharia e Pesquisa Aplicada*, v. 5, n. 1, p. 67–77, 2020. Citado na página 17.

SALTON, G.; BUCKLEY, C. Term-weighting approaches in automatic text retrieval. *Information processing & management*, Elsevier, v. 24, n. 5, p. 513–523, 1988. Citado 2 vezes nas páginas 26 e 48.

SCHÜTZE, H.; MANNING, C. D.; RAGHAVAN, P. *Introduction to information retrieval*. [S.l.]: Cambridge University Press Cambridge, 2008. v. 39. Citado 2 vezes nas páginas 20 e 25.

SILVA, L. H. S. e; ZUCOLOTO, G. F.; BARBOSA, D. B. de. Litigância predatória no brasil. *Radar: Tecnologia, Produção e Comércio Exterior*, n. 22, 2013. Acessado em: 5 jan. 2025. Disponível em: <https://repositorio.ipea.gov.br/handle/11058/6796>. Citado na página 13.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Bertimbau: pretrained bert models for brazilian portuguese. In: SPRINGER. *Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I 9*. [S.l.], 2020. p. 403–417. Citado na página 46.

VASWANI, A. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. Citado 3 vezes nas páginas 21, 22 e 23.

WIRTH, R.; HIPP, J. Crisp-dm: Towards a standard process model for data mining. In: MANCHESTER. *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*. [S.l.], 2000. v. 1, p. 29–39. Citado na página 37.

ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. *Dive into deep learning*. [S.l.]: Cambridge University Press, 2023. Citado na página 21.

ZHANG, H.; LIANG, H.; ZHAN, L.; LAM, A.; WU, X.-M. Revisit few-shot intent classification with plms: Direct fine-tuning vs. continual pre-training. *arXiv preprint arXiv:2306.05278*, 2023. Citado na página 22.