Universidade Federal do Paraná
Setor de Ciências Exatas
Departamento de Estatística
Programa de Especialização em *Data Science* e *Big Data*

Gustavo Ribeiro Dortas

Análise de similaridade em textos jurídicos: desenvolvimento de uma ferramenta de apoio à elaboração de votos para Acórdãos do Tribunal de Contas do Estado do Paraná

Curitiba
2024

Gustavo Ribeiro Dortas

Análise de similaridade em textos jurídicos: desenvolvimento de uma ferramenta de apoio à elaboração de votos para Acórdãos do Tribunal de Contas do Estado do Paraná

Monografia apresentada ao Programa de Especialização em *Data Science* e *Big Data* da Universidade Federal do Paraná como requisito parcial para a obtenção do grau de especialista.

Orientador: Prof. Walmes Marques Zeviani.

Curitiba
2024

Análise de similaridade em textos jurídicos

Desenvolvimento de uma ferramenta de apoio à elaboração de votos para Acórdãos do Tribunal de Contas do Estado do Paraná

Gustavo R. Dortas[1], Walmes M. Zeviani[2]

[1]Aluno do programa de Especialização em Data Science & Big Data. Auditor de Controle Externo - TCE/PR*
[2]Professor do Departamento de Estatística - DEST/UFPR†

O presente trabalho apresenta uma análise de similaridade em textos jurídicos com o objetivo de desenvolver uma ferramenta de apoio à elaboração de votos para Acórdãos do Tribunal de Contas do Estado do Paraná (TCE-PR). Utilizando tecnologias de processamento de linguagem natural e análise de dados, a pesquisa propõe uma ferramenta de inteligência artificial que pode analisar, comparar e recuperar informações de acórdãos anteriores de maneira rápida e precisa. Foram avaliados três algoritmos: BM25, TF-IDF e Gensim, com o BM25 se destacando por sua eficiência em identificar similaridades semânticas relevantes. Conclui-se que a utilização de tecnologias de processamento de linguagem natural (NLP), em especial o algoritmo BM25, se mostrou eficiente na pesquisa de jurisprudência no Tribunal de Contas do Estado do Paraná (TCE-PR).
Palavras-chave: Análise de Similaridade, Textos Jurídicos, Processamento de Linguagem Natural, Inteligência Artificial, Tribunal de Contas do Estado do Paraná (TCE-PR)

The present work presents a similarity analysis in legal texts aiming to develop a support tool for the preparation of votes for Judgments of the Court of Accounts of the State of Paraná (TCE-PR). Using natural language processing technologies and data analysis, the research proposes an artificial intelligence tool capable of analyzing, comparing, and retrieving information from previous judgments quickly and accurately. Three algorithms were evaluated: BM25, TF-IDF, and Gensim, with BM25 standing out for its efficiency in identifying relevant semantic similarities. It is concluded that the use of natural language processing (NLP) technologies, especially the BM25 algorithm, proved to be efficient in jurisprudence research at the Court of Accounts of the State of Paraná (TCE-PR).
Keywords: Similarity Analysis, Legal Texts, Natural Language Processing, Artificial Intelligence, Court of Accounts of the State of Paraná (TCE-PR)

1. Introdução

Nas rotinas dos gabinetes dos Conselheiros do Tribunal de Contas do Estado do Paraná (TCE-PR), a pesquisa e análise de acórdãos anteriores é uma atividade essencial. Esta pesquisa permite que os conselheiros e suas equipes revisitem decisões históricas, compreendam o raciocínio por trás desses julgamentos e apliquem esses conhecimentos a casos em andamento. Contudo, a grande quantidade de casos e a complexidade dos temas fazem dessa tarefa um grande desafio, exigindo tempo e dedicação. O acesso rápido e preciso a informações relevantes torna-se um ponto importante para a eficiência e eficácia do processo de elaboração dos votos.

Neste contexto, a presente pesquisa busca explorar o uso de tecnologias de processamento de linguagem natural e análise de dados para otimizar a pesquisa de jurisprudência no TCE-PR. A proposta é desenvolver uma ferramenta de inteligência artificial capaz de analisar, comparar e recuperar informações de acórdãos anteriores de forma rápida e precisa. Tal ferramenta não só economizaria um tempo significativo para os conselheiros e suas equipes, como também aumentaria a precisão e a profundidade nas pesquisas.

1.1. Contextualização

O Tribunal de Contas do Estado do Paraná (TCE-PR) é uma instituição fundamental no controle externo e na fiscalização da administração pública estadual e municipal. Ele é responsável por avaliar a legalidade, legi-

*dortas.gu@gmail.com
†walmes@ufpr.br

timidade e economicidade dos gastos públicos, além de julgar as contas dos administradores e demais responsáveis por dinheiros, bens e valores públicos. O trabalho do TCE-PR é essencial para garantir a correta aplicação dos recursos públicos e a transparência nas ações governamentais.

As deliberações do TCE-PR podem ser monocráticas ou realizadas por seus órgãos colegiados, compostos por Conselheiros encarregados de julgar as contas e processos que lhes são apresentados. Estas deliberações ocorrem em sessões plenárias ou nas câmaras, onde são discutidos e votados os processos. A deliberação mais comum e importante no contexto do TCE-PR é o Acórdão, que representa a decisão formal do Tribunal em um determinado caso.

Os acórdãos são documentos que registram as decisões dos órgãos colegiados do Tribunal. Nele, constam os aspectos legais e técnicos analisados, a síntese dos argumentos apresentados pelo Relator do processo, a decisão tomada pelos membros do colegiado e, quando aplicável, as recomendações ou sanções impostas.

1.2. Definição do Problema

O Tribunal de Contas do Estado do Paraná (TCE-PR) apresentou uma flutuação significativa no número de Acórdãos publicados nos últimos anos. Em 2019, foram registrados 4200 acórdãos. No entanto, nos anos subsequentes - 2020, 2021 e 2022 - houve uma redução nesses números, provavelmente como reflexo dos impactos gerados pela pandemia da COVID-19, que afetou diversas esferas da administração pública e da sociedade como um todo. Esta tendência, contudo, reverteu-se em 2023, com um total de 3931 acórdãos publicados, indicando um novo aumento na quantidade de casos processados e julgados pelo tribunal. Este padrão de flutuação evidencia não apenas as variações na demanda de trabalho do TCE-PR, mas também ressalta a necessidade de mecanismos eficientes para gerenciar e analisar tal volume de informações.

Muitos dos processos julgados pelo TCE-PR tratam de assuntos recorrentes, onde temas já pacificados no entendimento jurídico são frequentemente revisitados. Isso resulta em uma situação em que os Conselheiros e suas equipes se deparam com a necessidade de elaborar votos em casos que, embora distintos em seus detalhes, são similares em sua natureza e contexto jurídico. Não é raro que votos muito semelhantes sejam redigidos para casos parecidos, evidenciando

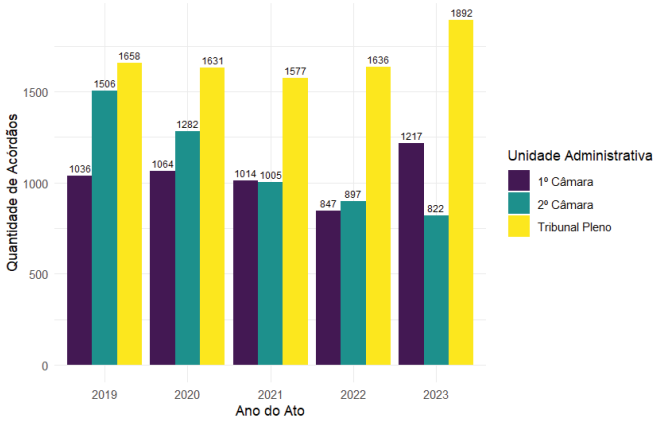

Figura 1: Quantidade de Acórdãos publicados por Unidade Administrativa.

uma oportunidade significativa de otimização desse processo.

A introdução de algoritmos de análise de similaridade pode desempenhar um papel importante na etapa de confecção de um voto do relator. Estes algoritmos têm a capacidade de identificar decisões anteriores que são relevantes para um caso atual, permitindo uma revisão rápida e eficiente de jurisprudências e votos anteriores. Além disso, a possibilidade de buscar decisões similares, sejam elas de autoria do próprio Conselheiro em questão ou de seus pares, fornece uma base mais ampla para a compreensão de diferentes posicionamentos jurídicos.

2. Fundamentação Teórica

Neste capítulo, será abordado conceitos e metodologias que fundamentam a análise de similaridade semântica entre textos. Essas técnicas são essenciais para compreender como as palavras podem ser matematicamente quantificadas e comparadas em diversos contextos.

Segundo (Manning, et al., 1999) [7], cada palavra pode ser representada como um vetor em um espaço multidimensional, onde cada dimensão pode indicar uma característica específica da palavra, como sua frequência e presença em diversos documentos.

A representação de palavras como vetores em espaços multidimensionais permite que as relações semânticas sejam capturadas de forma mais eficiente, possibilitando que modelos computacionais processem linguagem natural de maneira que simule a compreensão humana.

O gráfico de dispersão t-SNE apresentado na **Figura 2** ilustra a distribuição de dez frases distintas em um espaço bidimensional, resultante do processo de redução de dimensionalidade dos embeddings obtidos pelo modelo de linguagem BERT. Cada ponto no gráfico representa uma frase, e a proximidade entre os pontos reflete a similaridade semântica entre elas, conforme capturada pelo modelo BERT e projetada pelo t-SNE.

Podemos observar agrupamentos de pontos que indicam a semelhança temática entre as frases. Por exemplo, as frases "O céu está azul hoje."e "O sol brilha no céu claro."estão próximas no gráfico, sugerindo que compartilham um contexto semântico similar relacionado ao clima e ao céu. Da mesma forma, "Está chovendo bastante agora."e "A chuva continua durante a tarde."formam outro grupo, indicando uma conexão semântica relacionada ao fenômeno da chuva.

Frases sobre tópicos distintos, como "Os gatos gostam de dormir."e "Estou aprendendo a programar em Python.", estão mais distantes umas das outras, refletindo uma menor similaridade semântica. O posicionamento dessas frases no espaço bidimensional demonstra a capacidade dos modelos analise de similaridade em extrair e representar relações semânticas em determinados textos.

As metodologias de análise de similaridade têm suas particularidades e níveis de complexidade, mas todas buscam um mesmo fim: converter escrita em dados numéricos que permitem verificar o quanto um conteúdo se assemelha ao outro. A seguir apresentaremos 4 (quatro) metodologias.

2.1. Bag of Words (BoW)

No modelo Bag of Words (BoW), cada documento é representado por um vetor onde cada dimensão corresponde a uma palavra específica do vocabulário. A presença e frequência das palavras no documento são enfatizadas, enquanto a ordem e a estrutura gramatical são ignoradas. Isso resulta em uma representação simplificada, mas eficaz, útil para várias aplicações de processamento de linguagem natural, como classificação de documentos e análise de sentimentos. A eficácia do BoW decorre de sua capacidade de transformar texto não estruturado em uma forma estruturada, facilitando a aplicação de algoritmos de aprendizado de máquina.

2.2. Term Frequency - Inverse Document Frequency

O modelo Term Frequency - Inverse Document Frequency (TF-IDF) é baseado na combinação de dois conceitos principais: a frequência do termo e a frequência inversa do documento. Inicialmente, a frequência do termo (TF) é calculada pela contagem da ocorrência de uma palavra específica em um documento. Comumente, para evitar a distorção causada por altas frequências, utiliza-se o logaritmo da contagem. Esta abordagem parte do princípio de que uma palavra que aparece muitas vezes em um documento não necessariamente o torna 100 vezes mais relevante para o significado do documento (Jurafsky, et al., 2023) [2]. Assim, a frequência do termo é ajustada para refletir de maneira mais equilibrada a importância do termo dentro do documento.

2.3. Doc2Vec

O termo "Paragraph Vector"foi introduzido por (Le, et al., 2014) [3] onde propuseram uma técnica para superar as limitações dos modelos de bag of words, que ignoram a ordem e a semântica das palavras dentro dos textos. O algoritmo Paragraph Vector, também conhecido por sua implementação como Doc2Vec em algumas bibliotecas de Processamento de Linguagem Natural, cria vetores densos e de tamanho fixo para representar textos, treinando-os para prever palavras dentro de um contexto em um documento.

A principal inovação do Paragraph Vector/Doc2Vec é sua capacidade de capturar o contexto e a informação semântica dos textos, permitindo que o modelo entenda melhor a similaridade e a relação entre diferentes peças de texto. Essa abordagem é significativamente mais poderosa do que métodos anteriores, pois cada documento é representado por um vetor único no espaço vetorial, que encapsula as características essenciais do texto.

2.4. Best Matching 25 (BM25)

O algoritmo BM25 destaca-se como um dos modelos mais influentes e amplamente adotados para a tarefa de ranqueamento de documentos baseado em sua relevância para uma consulta específica.

Este algoritmo é centrado na ideia de que a relevância de um documento em relação a uma consulta pode ser modelada e quantificada probabilisticamente, considerando a presença e a distribuição de termos de consulta dentro do documento e no corpus como um todo (Robertson, et al., 2009) [5].

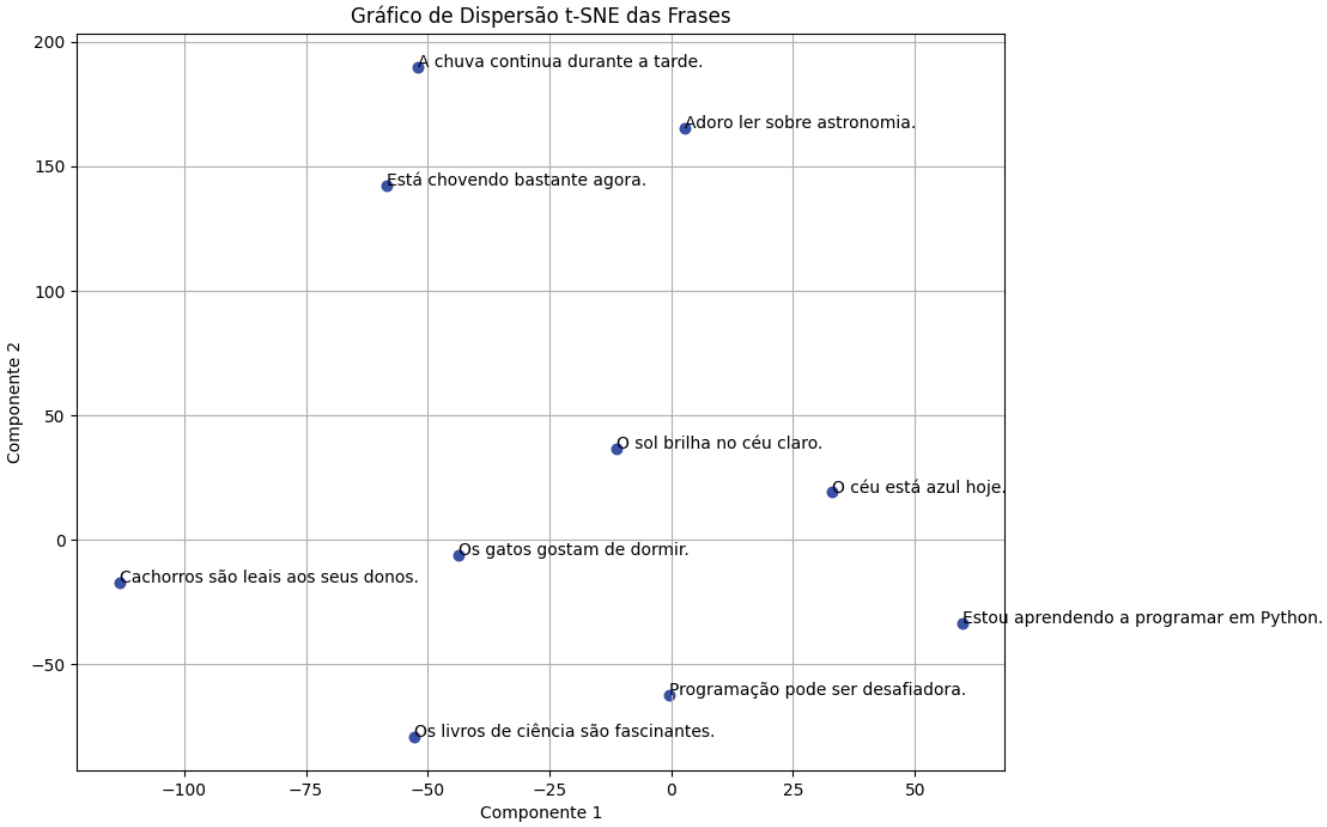

Figura 2: Gráfico de dispersão t-SNE.

O BM25, uma evolução dos modelos anteriores do PRF, como o BM11 e o BM15, incorpora melhorias significativas na forma como a frequência dos termos nos documentos (term frequency, TF) e a inversa da frequência dos documentos no corpus (inverse document frequency, IDF) são calculadas e combinadas. Essas melhorias permitem que o BM25 seja menos sensível a variações extremas na TF, equilibrando a importância entre termos frequentes e raros, o que resulta em um ranqueamento mais eficaz e relevante dos documentos.

Ademais, o BM25 introduz a noção de fator de normalização, que ajusta o escore de relevância de um documento com base no seu comprimento, mitigando assim o viés em favor de documentos longos ou curtos demais. Essa característica é particularmente relevante no domínio jurídico, onde os documentos podem variar significativamente em comprimento e complexidade, demandando um modelo que possa se adaptar a essas variações sem comprometer a precisão da recuperação da informação.

3. Trabalhos Relacionados

No contexto jurídico, onde a precisão e a interpretação de textos são fundamentais, a aplicação de tecnologias como a aprendizagem de máquina oferece um potencial significativo para inovação. Isso se deve ao crescente interesse em aproveitar dados textuais não estruturados presentes em documentos jurídicos, impulsionado pelos avanços em inteligência artificial.

A dissertação apresentada por (Bispo, 2022) [1] propõe uma metodologia para aprimorar a análise e a redação de sentenças judiciais no âmbito da Justiça do Trabalho, por meio do sistema Processo Judicial Eletrônico (PJe). O objetivo é possibilitar aos magistrados uma pesquisa mais eficiente por sentenças similares, incluindo a exibição do grau de similaridade entre os documentos encontrados e o caso em análise.

Utilizando técnicas de mineração de texto, o estudo avalia a eficácia de modelos computacionais distintos - Latent Dirichlet Allocation (LDA), Doc2Vec e Best Match 25 (BM25) - na identificação de similaridades entre sentenças. Os modelos são testados com base em um conjunto de dados composto por sentenças do Tribunal Regional do Trabalho da 10ª Região, tendo

o BM25 demonstrado superioridade em performance em relação aos demais, segundo métricas de Precisão aos K (P@K) e Ganho Cumulativo Descontado Normalizado (nDCG). A integração do modelo BM25 ao PJe visa oferecer um recurso de apoio à decisão judicial, facilitando a identificação de precedentes e contribuindo para a consistência das sentenças emitidas. O trabalho destaca a importância da aplicação de métodos de recuperação de informação e análise semântica de textos jurídicos, visando a otimização dos processos judiciais e a promoção de uma maior eficiência na Justiça do Trabalho.

A pesquisa feita por (Menon, et al., 2021) [4] apresenta uma solução computacional para a classificação e predição de sentenças judiciais, aplicando técnicas de Inteligência Artificial (IA) no âmbito do Direito. O foco da pesquisa foi a classificação de petições iniciais de ações civis públicas de improbidade administrativa e de execuções de termos de ajustamento de conduta, propostas pelo Ministério Público do Paraná, entre 2011 e 2018. Utilizando-se de métodos de processamento de linguagem natural e aprendizagem de máquina, o estudo alcançou uma acurácia de 78,02% com o uso do algoritmo Logistic Regression, a técnica Bag of Words para representação dos dados, e a stemização RSLPS para a redução da dimensionalidade dos textos.

A análise de sentenças judiciais enfrentou desafios significativos, como a heterogeneidade e a complexa formatação dos documentos jurídicos, o desbalanceamento entre as classes de sentenças, a alta dimensionalidade e diversidade dos dados devido à especificidade e à riqueza da linguagem jurídica, bem como a seleção de técnicas de pré-processamento e representação de dados que se adequarem ao vocabulário e à estrutura única do discurso jurídico. Adicionalmente, a presença de ruídos nos textos, como cabeçalhos e referências legais, e a sensibilidade dos algoritmos a esses elementos não textuais, representaram obstáculos adicionais, destacando a necessidade de métodos de limpeza, seleção e normalização dos dados para garantir a eficácia e a aplicabilidade das soluções de IA no campo do Direito.

Em (Silva, et al., 2021) [6], foi apresentado uma solução para a instrução assistida de pareceres sobre processos judiciais no Tribunal de Contas da União (TCU), chamado Assistente Conjur. Este sistema utiliza técnicas avançadas de Inteligência Artificial (IA), incluindo Processamento de Linguagem Natural (NLP), Machine Learning e Deep Learning, para automatizar e otimizar a criação de pareceres jurídicos sobre temas recorrentes. O objetivo principal do projeto foi substituir o método convencional de elaboração desses pareceres por um processo automatizado, acelerando significativamente a rotina de trabalho dos servidores do TCU.

Os resultados alcançados com a implementação do Assistente Conjur foram notáveis, com reduções significativas no tempo necessário para a elaboração de pareceres. A padronização e a confiabilidade dos pareceres foram aprimoradas, enquanto a atualização constante do banco de conhecimento assegura que os servidores tenham acesso às informações mais recentes e relevantes.

4. Material e Métodos

Neste capítulo, detalhamos a metodologia adotada, que envolve a coleta, processamento e análise de acórdãos do Tribunal de Contas do Estado do Paraná (TCE-PR).

4.1. Conjunto de Dados

O primeiro passo neste processo foi a extração dos acórdãos, realizada através de um método de web scraping utilizando a linguagem Python.

O processo começa com a configuração do Selenium WebDriver, uma ferramenta de automação de navegador web. O Selenium foi utilizado para para interagir com o site do TCE-PR, navegando pelas páginas e acessando os documentos desejados. Para isso, o WebDriver é configurado com opções específicas, como o tamanho da janela do navegador e o caminho para o driver do Chrome (chromedriver).

Após, estabeleceu-se uma conexão com um banco de dados do TCE-PR utilizando a biblioteca SQLAlchemy. Através da conexão com o banco de dados SQL Server do TCE/PR, uma consulta foi realizada para extrair informações de processos, tais como o número do processo, o assunto, o nome do relator e o nome do acórdão. Os dados obtidos foram então armazenados em um *D*ataFrame com a biblioteca Pandas.

Por fim, essas informações são registradas em um arquivo CSV.

4.2. Implementação do Modelo TF-IDF

A fase inicial do processo consiste na preparação e normalização dos dados textuais. Para tanto, faz-se uso da biblioteca *Pandas* para a manipulação e organização dos dados, e da biblioteca *N*LTK, que é amplamente

reconhecida por suas funcionalidades no âmbito do Processamento de Linguagem Natural.

O *DataFrame*, constituído especificamente pelos textos dos acórdãos, é importado do arquivo designado por meio da função `pd.read_csv`. A coluna *Content*, que contém os textos dos acórdãos, é submetida a um pré-processamento com a função `preprocess`, que executa as seguintes operações em cada texto:

1. **Tokenização e Remoção de Pontuação**: Esta etapa envolve a segmentação do texto em unidades menores (tokens) e a eliminação de caracteres de pontuação, utilizando-se, para isso, a função `word_tokenize` do NLTK e métodos de manipulação de strings do Python.
2. **Eliminação de Stopwords**: Palavras comuns que geralmente não contribuem para o significado semântico do texto são removidas. A lista de stopwords em português é fornecida pelo NLTK.
3. **Stemming**: Aplica-se a técnica de stemming, através do `RSLPStemmer` do NLTK, com o objetivo de reduzir as palavras a suas formas radicais, facilitando a normalização e a comparação textual.

Posteriormente, cada acórdão no *DataFrame* é processado conforme o modelo escolhido. Neste modelo, a representação numérica dos textos foi obtida por meio do modelo *Term Frequency-Inverse Document Frequency* (TF-IDF), que quantifica a importância de cada termo nos documentos em relação ao conjunto total de dados. Utiliza-se a classe `TfidfVectorizer` da biblioteca *scikit-learn* para construir a matriz TF-IDF, transformando os textos normalizados em vetores numéricos que refletem a relevância dos termos.

Para aferir a similaridade entre os documentos, foi utilizada a medida de similaridade do cosseno, que avalia o ângulo entre dois vetores no espaço vetorial criado pelo modelo TF-IDF. A função `calculate_similarity` processa uma entrada do usuário, converte-a em um vetor TF-IDF e calcula a similaridade desse vetor com todos os vetores da matriz TF-IDF dos textos dos acórdãos, por meio da função `cosine_similarity` da scikit-learn.

Os resultados foram filtrados para excluir aqueles com uma porcentagem de similaridade inferior a um limiar predefinido (X%), e são classificados em ordem decrescente de similaridade. Os documentos recomendados são então apresentados ao usuário, juntamente com a respectiva porcentagem de similaridade, indicando o grau de relação do conteúdo com a consulta realizada.

4.3. Implementação do Modelo BM25

Inicialmente, procedeu-se à coleta de dados, obtendo-se um conjunto de documentos em formato CSV, oriundos de registros de acórdãos consolidados. Na segunda etapa, o pré-processamento dos textos foi realizado com o objetivo de normalizar e preparar os dados para a aplicação do algoritmo *B*M25. Este processo incluiu a conversão de todos os textos para minúsculas, a remoção de caracteres não alfabéticos e especiais através de expressões regulares, e a tokenização, que consiste na divisão do texto em unidades básicas ou tokens. Estas etapas são importantes para reduzir a variabilidade dos dados e focar nos elementos textuais significativos para a análise.

Subsequentemente, a terceira etapa envolveu a eliminação de stopwords da língua portuguesa, palavras comuns que tendem a não contribuir para a relevância semântica do texto, como preposições, conjunções e artigos.

Com os dados devidamente pré-processados, a quarta etapa consistiu na construção de um índice inverso utilizando o algoritmo *B*M25Okapi, que permitiu a transformação dos textos processados em uma estrutura adequada para a recuperação e ranqueamento de informações. O BM25 é um modelo baseado em probabilidade que considera a frequência dos termos nos documentos e no corpus como um todo, bem como o comprimento dos documentos, para calcular os escores de relevância.

Finalmente, a aplicação prática do algoritmo foi testada através de uma consulta específica inserida no modelo *B*M25 construído. Esta etapa envolveu o cálculo dos escores de relevância para todos os documentos no índice em relação à consulta fornecida, seguida pela ordenação dos documentos de acordo com seus escores.

4.4. Implementação do Modelo Gensim

Foi utilizado o *pandas* para manipulação de dados, *re* para expressões regulares, *nltk* para ferramentas de PLN como tokenização e remoção de stopwords, e *gensim* para modelagem de tópicos e transformações de documentos. Além disso, *sklearn* foi usado para calcular a similaridade de cosseno entre vetores de documentos.

A função *preprocess* é definida para executar o pré-processamento dos textos. Este processo inclui a conversão para minúsculas, remoção de números, pontuações e caracteres especiais com o uso de expressões

regulares. Além disso, a tokenização é realizada para dividir o texto em palavras individuais, e as stopwords, palavras comuns que não contribuem significativamente para o significado do texto, são removidas.

Após o pré-processamento, foi construído um dicionário que mapeia cada palavra única a um índice numérico. Essa estrutura é imporante para a transformação dos textos em uma representação vetorial, que é feita através do método doc2bow do gensim. Esse método converte cada documento em um conjunto de tuplas, onde cada tupla contém o índice da palavra no dicionário e sua frequência no documento.

Em seguida, cria-se um modelo de Indexação Semântica Latente (LSI) a partir do corpus processado, especificando o número de tópicos (no presente estudo, 200). O parâmetro *num_topics* controla o número de dimensões latentes (tópicos) que o modelo deve considerar.

O LSI utiliza a técnica de decomposição em valores singulares (SVD) para reduzir a dimensionalidade do modelo, mantendo apenas as dimensões mais significativas que capturam a essência semântica dos dados. Esse processo permite uma melhor representação das relações semânticas entre as palavras, como sinônimos próximos ou conceitos distantes, otimizando o espaço de características para facilitar a recuperação e a análise de informações, minimizando distorções que seriam evidentes em um espaço com dimensionalidade inadequada.

Em (Landauer, et al., 2007) [8], é apresentado um exemplo elucidativo sobre a importância da redução de dimensionalidade, comparando a representação espacial de pontos geográficos em diferentes dimensões. No exemplo citado, medem-se as distâncias entre cidades como Oslo, Bagdá e Sydney, tentando-se representá-las primeiramente em uma linha reta (uma dimensão), o que resulta em distorções significativas e inadequações, pois não é possível acomodar a realidade tridimensional dos locais em um espaço unidimensional. Ao aumentar para duas dimensões, as distorções são reduzidas, mas ainda assim são significativas. Usando três dimensões, como num globo, a representação se torna muito mais precisa, embora simplificações como a ignorância das elevações ainda impliquem em certas reduções. Isso exemplifica como, ao aumentar as dimensões, a representação dos dados se torna mais fiel à realidade, mas até um certo ponto, além do qual aumentos adicionais em dimensionalidade (como mudar de milhas para pés) trazem pouca ou nenhuma melhoria perceptível. Esta analogia é usada para destacar a busca por um número ótimo de dimensões no LSI, onde cada dimensão adicionada deve contribuir significativamente para a precisão da análise semântica.

Finalmente, a representação vetorial pode ser usada para calcular a similaridade entre documentos. No contexto deste estudo, optamos pela similaridade de cosseno, que é uma métrica comum para medir o ângulo entre dois vetores no espaço multidimensional, refletindo o grau de semelhança em termos de conteúdo textual.

5. Resultados

Para a avaliação da performance dos modelos de análise de similaridade estudados no capítulo anterior, foram selecionados cinco inputs:

- I - Licitação aquisição de equipamentos hospitalares;
- II - Irregularidades merenda escolar;
- III - Contratos de manutenção predial;
- IV - Irregularidades despesas com pessoal;
- V - Auditorias educação infantil.

Cada um desses inputs foram submetidos aos modelos TF-IDF, BM25 e Gensim. O objetivo era processar e coletar os 10 primeiros resultados retornados por cada um deles, com a finalidade de avaliar e comparar a eficácia de cada modelo em capturar relações semânticas pertinentes ao input fornecido.

A avaliação da relevância semântica dos resultados obtidos foi realizada utilizando uma escala de pontuação de 1 a 5, onde 1 indica uma relação semântica fraca ou inexistente entre o resultado e o input, e 5 indica uma relação semântica muito forte.

Os totais de pontuação para cada modelo foram então comparados para determinar qual deles apresentou a melhor performance geral no conjunto de testes utilizado. Esses dados estão sintetizados na Tabela 1, que apresenta a pontuação total para cada modelo em relação a cada input.

Modelo	I	II	III	IV	V	Total Geral
BM25	32	42	32	45	42	193
TF-IDF	24	34	17	40	38	153
Gensim	19	33	14	38	36	140

Tabela 1: Pontuação total para cada modelo em relação ao input.

Os resultados indicam que o modelo BM25 obteve a maior pontuação total (193). Isso pode ser atribuído à

sua capacidade de ponderar a relevância dos termos, levando em consideração tanto a frequência dos termos quanto o comprimento dos documentos, proporcionando uma análise mais precisa das similaridades semânticas.

6. Conclusão

Esta pesquisa apresentou um estudo dentro do domínio de Recuperação da Informação (RI) para textos jurídicos, especificamente para acórdãos do Tribunal de Contas do Estado do Paraná (TCE-PR). A análise se concentrou na utilização de tecnologias de processamento de linguagem natural (NLP) para otimizar a pesquisa de jurisprudência. Avaliou-se o desempenho de três algoritmos para encontrar similaridades entre os acórdãos: BM25, TF-IDF e Gensim. Nesse contexto, foram selecionados cinco input's com temas comuns tratados nos Acórdãos do TCE/PR.

Portanto, depois da avaliação dos experimentos, foi possível concluir que a utilização de tecnologias de processamento de linguagem natural (NLP), em especial o algoritmo BM25, se mostrou eficiente na pesquisa de jurisprudência no Tribunal de Contas do Estado do Paraná (TCE-PR). O BM25 destacou-se por sua capacidade de identificar similaridades semânticas relevantes entre os acórdãos, permitindo uma recuperação rápida e precisa das informações necessárias para ajudar na elaboração de votos. Essa eficiência não apenas economiza tempo dos Conselheiros e suas equipes, mas também garante maior profundidade e consistência nas pesquisas, contribuindo para decisões fundamentadas e coerentes.

7. Trabalhos Futuros

Existem algumas direções promissoras para trabalhos futuros que podem expandir e aprimorar as implementações aqui apresentadas.

Uma linha de pesquisa futura poderia focar na integração de técnicas de aprendizado profundo (Deep Learning), como Transformers e modelos de linguagem pré-treinados (por exemplo, BERT e GPT), para melhorar ainda mais a precisão e a relevância das buscas de jurisprudência.

Outro caminho interessante seria a implementação de uma interface interativa baseada em inteligência artificial generativa. Com a evolução dos modelos de linguagem, como o GPT-3 e o GPT-4, há a possibilidade de não apenas encontrar similaridades entre acórdãos, mas também de gerar automaticamente esboços de votos. Esta ferramenta poderia ser utilizada para criar um primeiro rascunho de um voto baseado em precedentes identificados, economizando tempo e garantindo a consistência nas decisões.

Agradecimentos

Gostaria de expressar minha gratidão ao orientador, Prof. Walmes M. Zeviani, pelas contribuições nesta pesquisa. Agradeço também aos integrantes do laboratório de inovação do TCE/PR pela colaboração e pelo ambiente estimulante que me proporcionaram durante o desenvolvimento deste trabalho.

Referências

[1] G. D. Bispo, *Inferência de similaridade de sentenças judiciais na Justiça do Trabalho*, (2022).

[2] D. Jurafsky e J. H. Martin, *Speech and Language Processing*, (2023).

[3] Q. Le e T. Mikolov, *Distributed Representations of Sentences and Documents*, Proceedings of the 31st International Conference on Machine Learning, PMLR 32(2):1188-1196, (2014).

[4] L. T. Menon, et al., *Inteligência Artificial e Direito: Uma Solução Computacional Capaz de Prever Decisões Judiciais*, Revista Humanidades e Inovação, p. 47, (2021).

[5] S. Robertson e H. Zaragoza, *The Probabilistic Relevance Framework: BM25 and Beyond*, Now Publishers Inc, Vol. 3, No. 4, p. 333–389, (2009).

[6] L. A. D. e Silva, C. C. S. Stigert e L. A. da Silva Pacheco, *Instrução Assistida de Pareceres sobre Processos Judiciais: assistente Conjur*, Revista do TCU, Julho-Dezembro, (2021).

[7] C. D. Manning e H. Schutze. *Foundations of statistical natural language processing*, Massachusetts Institute of Technology (1999).

[8] T. K. Landauer, D. S. McNamara, S. Dennis e W. Kintsch *Handbook of latent semantic analysis*, Lawrence Erlbaum Associates Publishers (2007).