Universidade de Brasília
Departamento de Estatística

Similaridade de Processos Judiciários Utilizando Processamento de Linguagem Natural

Bruno Gondim Toledo

Relatório apresentado para o Departamento de Estatística da Universidade de Brasília como parte dos requisitos necessários para obtenção do grau de Bacharel em Estatística.

Brasília
2025

Bruno Gondim Toledo

Similaridade de Processos Judiciários Utilizando Processamento de Linguagem Natural

Orientadora: Profª Drª Thais Carvalho Valadares Rodrigues

Relatório apresentado para o Departamento de Estatística da Universidade de Brasília como parte dos requisitos necessários para obtenção do grau de Bacharel em Estatística.

Brasília
2025

All models are wrong, but some are useful.

(George E. P. Box)

Para minha mãe

Agradecimentos

Agradeço a todos que participaram da minha jornada até aqui como pessoa e estudante. Agradeço a minha família, notoriamente nas pessoas da minha mãe Carla, do meu pai Adriano, das minhas avós Odete e Maria (*in memorian*), do meu avô Nildo (*in memorian*), meus irmãos Pablo e Taissa, demais tios, tias, primos, primas e do meu tio Alei e prima Cecília pelo auxílio para ingresso no curso. Agradeço aos meus amigos que, mesmo sem conhecimento ou interesse por estatística, sempre estiveram disponíveis para me auxiliar e ouvir reclamações e desesperos no decorrer do curso, nas pessoas de Daniel, Uirá, Bruno, Rosa, Natália, Lucca e Guilherme. Agradeço aos meus colegas e amigos de curso, com quem pude sempre contar e dividir experiências do curso, nas pessoas do Rafael, Micael, Kevyn, César, Davi, Arthur, Breno, Júlia e Eduardo. Agradeço aos meus professores, na pessoa da minha orientadora, a Profª Drª Thais Carvalho Valadares Rodrigues, por todo o conhecimento passado e todos os auxílios prestados. Aos meus supervisores de estágio Pamella e Euler, pela oportunidade da tradução dos conhecimentos teóricos obtidos no curso em experiências práticas — além da proposta deste tema para minha monografia — em um ambiente de tanta relevância quanto o Supremo Tribunal Federal do Brasil, bem como aos colegas de STF Lucas, Nilsen e Fernando por todo o auxílio prestado na obtenção dos insumos necessários para este trabalho. Agradeço aos profissionais de saúde brasileiros por viabilizar uma existência durante e após a pandemia de COVID-19. E, por fim, agradeço a minha namorada e companheira Julia, que sem dúvidas conhecê-la foi a melhor coisa que poderia ter me acontecido, com quem desejo partilhar uma vida.

Não teria como citar todas as pessoas importantes para mim em uma página. Vocês estiveram comigo nos meus melhores e piores momentos, e são o que compõem o sentido da minha vida e trabalho. Muito obrigado.

Resumo

Com objetivo de contribuir com uma das metas de gestão da presidência do Ministro Luís Roberto Barroso de diminuição do acervo de processos de controle concentrado de constitucionalidade do Supremo Tribunal Federal (STF), nesse trabalho foram aplicadas técnicas de processamento de linguagem natural para encontrar processos semelhantes no acervo. Foram utilizadas técnicas de vetorização, como *tokenização* e *Bag of Words*, bem como métricas de similaridade, como distância do cosseno e divergência de Jensen-Shannon, a fim de encontrar semelhanças entre uma petição de entrada e os processos em tramitação no STF. Nesse trabalho também foi desenvolvido um aplicativo *Shiny* para retornar os resultados da busca por processos similares, utilizando técnicas amplamente aceitas na bibliografia e em conformidade com demais tecnologias do Tribunal sempre que possível, a fim de legar uma ferramenta útil e prática aos analistas do Tribunal. Este estudo foi capaz de produzir uma aplicação com bom desempenho e baixo custo computacional utilizando um modelo de *Bag of Words* metrificados sobre a distância do cosseno entre os dados, e, portanto, pode ser uma ferramenta útil em auxiliar no cumprimento desta meta de gestão do Ministro Presidente.

Palavras-chave: **Agrupamento, Controle Concentrado de Constitucionalidade, Distância do Cosseno, Supremo Tribunal Federal, Vetorização de Texto.**

Abstract

With the aim of addressing one of the management goals of the presidency of Minister Luís Roberto Barroso, which is to reduce the backlog of concentrated constitutional Control pending cases at the Supreme Federal Court (STF), this study applied natural language processing techniques to identify similar cases among the pending cases at the Court. Techniques such as vectorization, including tokenization and Bag of Words, as well as similarity metrics like cosine distance and Jensen-Shannon divergence, were employed to find similarities between an incoming petition and the ongoing cases at the STF. Additionally, a Shiny application was developed to return the results of the search for similar cases, utilizing widely accepted techniques in the literature and aligning with the Court's existing technologies whenever possible, in order to provide a useful and practical tool for the Court's analysts. This study successfully produced an application with good performance and low computational cost by using a Bag of Words model measured by cosine distance between the data, and thus, it can be a valuable tool in assisting the achievement of this management goal set by the President Minister.

Keywords: **Clustering, Concentrated Constitutional Control, Cosine Distance, Supreme Federal Court, Text Vectorization.**

Lista de Tabelas

1 Integridade das Amostras 32

2 Similaridades entre ADI6931 e ADI6921. 37

3 Tabela de palavras mais frequentes 37

4 Similaridades entre ADPF857 e ADPF743, ADPF746. 38

5 Tabela de palavras mais frequentes 38

6 Similaridades entre ADO54 e ADPF760. 39

7 Tabela de palavras mais frequentes 39

8 Similaridades entre ADI7078 e ADI7066, ADI7070. 40

9 Tabela de palavras mais frequentes 40

10 Similaridades entre ADI3318 e ADI2943, ADI3309. 41

11 Tabela de palavras mais frequentes 41

Lista de Figuras

1 Distância Euclidiana entre os pontos **P** e **Q** 18

2 Distância de quarteirão entre os pontos $\mathbf{X}_i$ e $\mathbf{X}_k$. 19

3 O cosseno como medida de similaridade entre **A** e **B** 20

4 A observação x_0 sendo classificado segundo seus K vizinhos mais próximos 24

5 Exemplo simples de um dendograma 26

6 O algoritmo t-SNE aplicado aos dados MNIST. 27

7 Exemplo de nuvem de palavras 30

8 t-SNE do processo ADI5642 no acervo. 43

9 t-SNE do processo ADI7584 no acervo. 43

10 t-SNE do processo ADI7585 no acervo. 44

11 A aplicação ao ser executada. 45

12 Navegação pelos filtros da aplicação. 46

13 Retorno da aplicação após selecionar filtros e clicar em Atualizar. 47

14 Nuvens de palavras após clicar em caixas de seleção. 48

15 Segunda aba da aplicação — lista de dados. 48

16 Terceira aba da aplicação — palavras mais frequentes. 49

17 Quarta aba da aplicação — Interseções mais frequentes. 50

18 Forma dos dados exportados da aplicação. 51

19 Colagem de *print* da aplicação com *print* de informação à sociedade. . . . 52

Sumário

1 Introdução 12

2 Referencial Teórico 14

2.1 Pré-processamento 14

2.2 Vetorização 15

2.3 Medidas de similaridade e dissimilaridade 17

2.3.1 Distância Euclidiana 17

2.3.2 Distância de quarteirão 18

2.3.3 Distância de Mahalanobis 19

2.3.4 Distância de cossenos 20

2.3.5 Divergência de Jensen-Shannon 21

2.3.6 Índice de Tversky 21

2.3.7 Interseção Mínima de Probabilidade Empírica 22

2.4 Métodos de agrupamento 23

2.4.1 Vizinho mais próximo (KNN) 23

2.4.2 k-médias 24

2.5 Visualização de dados 25

2.5.1 Dendograma 25

2.5.2 t-SNE 26

2.5.3 Nuvens de palavras 29

3 Metodologia 31

3.1 Conjunto de dados 31

3.2 Métodos 32

4 Resultados 36

4.1 Vetorização dos textos 36

4.2 Estudos de caso 37

4.2.1 Estudo de caso 1 37

4.2.2 Estudo de caso 2 38
4.2.3 Estudo de caso 3 39
4.2.4 Estudo de caso 4 39
4.2.5 Estudo de caso 5 40
4.2.6 Análise 42
4.2.7 t-SNE 42
4.3 Aplicação 45
5 Conclusão 54
Referências 55

Abreviações e Siglas

- STF: Supremo Tribunal Federal
- STI: Secretaria de Tecnologia da Informação — STF
- NLP: *Natural Language Processing*
- PDF: *Portable Document Format*
- SQL: *Structured Query Language*
- OCR: *Optical Character Recognition*
- IMPE: Interseção mínima de probabilidade empírica
- KNN: k-nearest neighbors

1 Introdução

Modelagem estatística tornou-se essencial em diversos setores, inclusive no Poder Judiciário, visto que tem sido do interesse dos próprios tribunais saber indicadores e estatísticas relacionadas ao próprio tribunal, além de agilizar processos e direcionar os recursos humanos para tarefas menos mecânicas. A modelagem surge neste contexto para agregar valor às decisões estratégicas deste poder. Com o advento de novas técnicas computacionais, é possível agrupar e classificar textos do contexto jurídico, conforme descrito por Freitas et al. (2024).

O propósito deste trabalho é estudar semelhanças em processos de controle concentrado de constitucionalidade do Supremo Tribunal Federal (STF). Compõem o controle concentrado as seguintes classes processuais: Ações Diretas de Inconstitucionalidade (ADIs), as Ações Declaratórias de Constitucionalidade (ADCs), as Ações Diretas de Inconstitucionalidade por Omissão (ADOs) e as Arguições de Descumprimento de Preceito Fundamental (ADPFs). Em geral, estes processos são bastante diferentes uns dos outros, tratando de diversos temas que concernem ao direito constitucional brasileiro. Entretanto, é entendido pelo Tribunal que determinados assuntos correntes podem ocasionar no peticionamento de um ou mais processos de temas similares. A possível existência destes processos potencialmente similares motivam a busca no acervo vigente por processos que sejam semelhantes a um processo novo no Tribunal (petição inicial), a fim de encontrar alguns possíveis padrões que facilitem o encaminhamento processual destes processos semelhantes.

Com as petições iniciais dos processos de controle concentrado, foram aplicadas técnicas de vetorização (do inglês, *embedding*), distâncias estatísticas, técnicas de redução de dimensionalidade e métricas de comparação de similaridade. A escolha da petição inicial como fonte de dados para esta aplicação dá-se por dois motivos principais: é o primeiro documento recebido pelo Tribunal que trata do processo, possibilitando assim a busca destes processos semelhantes entre si antes de qualquer avaliação, marcação ou tramitação do processo, e também por ser o insumo atualmente utilizado pelos analistas do Tribunal para busca manual destes processos potencialmente semelhantes. Para utilização de modelos de vetorização do texto, realizou-se uma etapa anterior de pré-processamento do texto obtido, como a remoção de *stopwords*, pontuações, termos legais, números, espaços em branco, conforme discutido em Freitas et al. (2023).

Por conta da natureza cíclica dos processos que compõem o acervo do STF, não é esperado a formulação de classes rígidas para agrupamento, sendo somente necessário

encontrar similaridades e dissimilaridades entre os processos em atual tramitação a fim de agrupá-los de forma a facilitar o fluxo de trabalho do tribunal. Por exemplo, ao agrupar processos semelhantes, estes podem ser analisados conjuntamente, facilitando o fluxo de trabalho do STF. Além disso, a identificação de processos semelhantes pode facilitar uma priorização de pauta em alguns casos. Portanto, este trabalho propôs-se a estudar técnicas de similaridade destes processos de forma funcional e contínua, fornecendo assim um produto para o tribunal, a fim de ser utilizado pelos analistas que hoje buscam manualmente essas similaridades.

Portanto, o objetivo desse trabalho é encontrar os processos em tramitação mais semelhantes a um dado processo de referência (petição inicial), a fim de fornecer subsídios aos responsáveis pelo encaminhamento dos processos que chegam ao STF, visando reduzir o trabalho mecânico dos servidores do tribunal que atualmente necessitam analisar e encaminhar processos semelhantes aos setores responsáveis.

Para viabilizar este objetivo, o procedimento adotado no decorrer deste trabalho envolveu as seguintes etapas:

- Processamento dos dados empregando técnicas de Processamento de Linguagem Natural (NLP), transformando as petições iniciais dos processos em PDF (dados de entrada) por diferentes técnicas;
- Comparação da efetividade destas técnicas em encontrar processos semelhantes;
- Construção de um aplicativo *Shiny* para efetiva utilização deste produto pelo Tribunal.

Este trabalho está organizado da seguinte forma: no capítulo 2, Referencial Teórico, estão definidas as técnicas utilizadas para metrificação de semelhanças entre os processos. No capítulo 3, Metodologia, está descrita a forma da utilização destas técnicas no escopo do trabalho. No capítulo 4, Resultados, estão listados os resultados da aplicação destas técnicas aos dados, bem como a apresentação do aplicativo *Shiny* desenvolvido para a efetiva utilização destas técnicas na rotina de trabalho. Por fim, no capítulo 5, Conclusão, encontra-se a conclusão dos resultados obtidos.

2 Referencial Teórico

O referencial aplicado para processamento e conversão de dados textuais em vetores numéricos (vetorização) para posterior metrificação de similaridades foi análogo ao aplicado por Freitas et al. (2024), consistindo nas etapas de limpeza de dados, vetorização e organização dos dados. Após o processamento inicial e modelagem, os dados foram comparados segundo medidas de similaridade. Buscou-se também metodologias mais simples, como comparação dos corpos textuais e *Bag of Words* das petições utilizando medidas de distância estatística. Assim, as principais técnicas de pré-processamento, medidas de similaridade e visualização de dados serão descritas nesse capítulo.

2.1 Pré-processamento

Rotinas de pré-processamento devem ser realizadas para viabilizar qualquer aplicação de NLP, como em Soares (2022). No contexto deste trabalho, as petições iniciais disponibilizada pelo Portal do STF encontram-se em formato PDF. Dentre estas, observou-se petições oriundas de documentos impressos e posteriormente digitalizados e salvos neste formato, bem como documentos nativamente digitados em alguma plataforma e posteriormente exportados em formato PDF. Em ambos os casos, os documentos contêm imagens, figuras, timbres e demais elementos visuais que não compõem o paradigma desta aplicação e, portanto, necessitam ser filtrados. Funções implementadas em linguagens de programação (como *pdf_text* do pacote *pdftools* da linguagem de programação **R**) em geral já realizam a leitura do conteúdo textual do documento sem levar em consideração os elementos gráficos deste, para o caso de documentos nativamente digitais.

No caso dos documentos digitalizados, é necessária uma etapa posterior de "ocerização"(do acrônimo OCR - *Optical Character Recognition*). Esta técnica, também chamada de Reconhecimento ótico de caracteres, é utilizada para transcrever o conteúdo textual do documento — que na prática, é uma imagem — para caracteres numéricos e alfanuméricos. Diversas implementações de ocerizadores estão disponíveis na maior parte das linguagens de programação mais populares, muitas vezes na forma de pacotes. Para o escopo deste trabalho, em que o recorte amostral é do ano de 2016 até 2024, grande parte das petições iniciais encontravam-se disponíveis em PDF oriundos de documentos nativamente digitais, de modo que dispensavam uma etapa de ocerização. No caso das petições que necessitavam de ocerização, estas foram fornecidas diretamente pela STI do STF já ocerizadas pela implementação própria do tribunal desta tecnologia.

Tendo sido os textos importados para o ambiente de uma linguagem de programação, por vezes são necessários ajustes ortográficos das palavras importadas. Como nas tipografias mais usuais existem caracteres similares entre si (como a letra i maiúscula e a letra l minúscula, ou a letra o e o número 0), ou ainda palavras que sofrem translineação (isto é, quando a palavra é dividida por um hífen quando não cabe em apenas uma linha), por vezes os algoritmos utilizados nas etapas anteriores de pré processamento retornam palavras incompletas ou sem sentido por conta desses fenômenos. Diversas técnicas podem ser utilizadas para corrigir estas distorções, como o algoritmo de Metropolis-Hastings aplicado à descriptografia, utilizado por Veroutis e Fajardo (2021). Para os dados desta aplicação, foi realizada uma rotina para correção destas distorções nos dados pela STI do STF. Detalhes do funcionamento desta tecnologia não podem ser descritas por motivos de segurança de informação.

Sob a intenção de aplicar técnicas de vetorização, o estado da arte de NLP sugere que sejam removidas palavras que não contribuem informativamente com a semântica de um texto, conforme descrito por Sarica e Luo (2021). Em geral, constituem as ditas palavras de parada (ou ainda palavras vazias, no inglês *stopwords*), conectivos e conjunções. Não existe consenso de quais palavras podem ser consideradas palavras vazias, posto que variam em função do contexto em que se pretende manipular estes textos. Desta forma, esta lista pode variar de centenas de palavras ou até nenhuma palavra — ver Ross (2020) e Nunes (2019) — a depender da aplicação.

2.2 Vetorização

Vetorização, ou *Embedding*, é uma técnica geralmente aplicada na área de processamento de linguagem natural. Em poucas palavras, é um procedimento de transformar texto em um vetor numérico.

A ideia por trás da vetorização é transformar os dados originais, texto, em um vetor numérico que mapeie e preserve a carga semântica do texto original. Sem embargos, é possível rapidamente identificar a utilidade desta técnica — visto que, para utilização de técnicas e algoritmos estatísticos, em geral buscamos dados numéricos, e não textuais — quando intencionamos aplicar alguma técnica para modelagem destes dados. Diversas técnicas para este procedimento vem sendo estudadas, implementadas e utilizadas, como por exemplo os métodos *Bag of Words* (Qader, Ameen e Ahmed, 2019), *Term Frequency — Inverse Document Frequency* (TF-IDF) (Salton e Buckley, 1988, apud Freitas et al, 2024), *Word2vec* (Mikolov et al, 2013, apud Freitas et al, 2024) e *doc2vec* (Le e Mikolov,

2014, apud Freitas et al, 2024).

Na técnica de *Bag of Words*, consideraremos o texto como "um conjunto desordenado de palavras, com sua posição no documento ignorada, mantendo apenas a sua frequência absoluta"(Jurafsky e Martin, 2025, p.58, tradução própria[1]). Ou seja, nesta técnica de vetorização, estaremos desconsiderando completamente o contexto de utilização de um termo, e estaremos preocupados simplesmente em contar as frequências de cada um dos termos presentes no texto. Desta forma, declarações simples como "isto causa aquilo"e "aquilo causa isto"seriam identificadas como idênticas nesta vetorização. Apesar da simplicidade e aparente fragilidade da técnica, esta vetorização pode ser útil e poderosa quando analisamos um texto mais longo, onde a contextualização local de um termo em um fragmento não seja tão relevante quanto a sua presença global no documento.

Métodos como *Word2Vec* e *doc2vec* buscam a obtenção do vetor numérico representando o texto original via treinamento de uma rede neural. No modelo *doc2vec*, por exemplo, os parâmetros do modelo (ou pesos) são atualizados buscando prever a próxima palavra dado o seu contexto (palavras ao redor) e dado um vetor de palavras que formam coletivamente o texto, conforme descrito em Freitas et al. (2024).

A utilização do contexto para a construção da vetorização pode aumentar a precisão desta. "O texto [...] é considerado parte integrante do contexto, e deve ser observado em relação ăs demais partes consideradas relevantes na declaração do contexto."(Firth, 1962, p.7, tradução própria[2]). Podemos argumentar que, ao considerar a vizinhança das palavras, conseguimos extrair significados que tragam sentido à utilização destas, inclusive aplicado ao contexto jurídico, evitando assim a homonímia. Por exemplo, um suposto processo com origem no estado do Rio de Janeiro que cite o nome do estado em sua petição inicial — suposição esta não descolada da realidade dos fatos — poderia confundir alguma técnica de vetorização que não considerasse vizinhanças para entendimento semântico do texto, visto que neste caso a palavra Rio não se refere ao corpo de ăgua, tampouco a palavra Janeiro se refere a um mês do calendário.

[1]No original: "an unordered set of words with their position ignored, keeping only their frequency in the document".

[2]No original: "The text in the focus of attention on renewal of connection with an instance, is regarded as an integral part of the context, and is observed in relation to the other parts regarded as relevant in the statement of the context".

2.3 Medidas de similaridade e dissimilaridade

Para agrupar os processos, estudou-se medidas de similaridade e dissimilaridade. Artes e Barroso (2023) argumentam que, para medidas de similaridade, quanto maior o valor, maior será a semelhança entre objetos, enquanto que para medidas de dissimilaridade, quanto maior o valor, mais diferente serão os objetos.

Uma medida de distância estatística pode ser definida simplesmente como a distância entre um ponto arbitrário P e um segundo ponto arbitrário Q, em qualquer espaço n-dimensional. Conforme definido por Johnson e Wichern (2007), qualquer medida de distância $d(P, Q)$ entre os pontos P e Q é válida se atender as seguintes propriedades, sendo R um terceiro ponto intermediário qualquer:

- $d(P, Q) = d(Q, P)$;
- $d(P, Q) > 0 \; se \; P \neq Q$;
- $d(P, Q) = 0 \; se \; P = Q$;
- $d(P, Q) \leq d(P, R) + d(R, Q)$ (Desigualdade triangular).

A última propriedade será evocada quando definirmos a distância do cosseno (seção 2.3.4). A seguir serão apresentadas algumas medidas de distância que podem ser exploradas, sendo elas as distâncias euclidiana, de quarteirão, de Mahalanobis e de cossenos, além da desigualdade de Jensen-Shannon e uma métrica própria criada com intenção de competir com as métricas já previamente estabelecidas, sendo esta uma forma modificada do índice de Tversky.

2.3.1 Distância Euclidiana

A distância Euclidiana é uma consequência direta do Teorema de Pitágoras. Segundo Johnson e Wichern (2007), considerando os pontos arbitrários P com coordenadas $P = (x_1, x_2, ..., x_p)$, e Q com coordenadas $Q = (y_1, y_2, ..., y_p)$, de uma hiperesfera p-dimensional, a distância reta entre eles, ou distância Euclidiana, é dada por:

$$d(P, Q) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_p - y_p)^2}. \tag{2.3.1}$$

A Figura 1 ilustra geometricamente a distância entre dois pontos **P** e **Q** no $\mathbb{R}^2$.

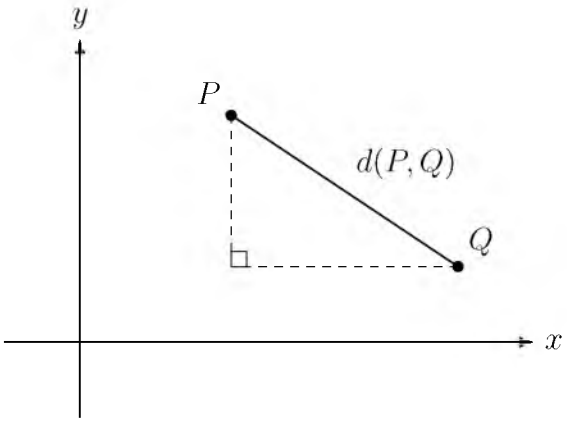

Figura 1: Distância Euclidiana entre os pontos **P** e **Q**

Fonte: elaboração própria.

Note que esta definição para P, Q pode ser expandida para a situação em que estes são vetores X_i, X_k, i, k amostras com p dimensões. Utilizando a notação de Artes e Barroso (2023):

$$d_{ik} = \sqrt{(x_i - x_k)^T (x_i - x_k)} = \sqrt{\sum_{j=1}^{p} (X_{ij} - X_{kj})^2}. \tag{2.3.2}$$

A distância euclidiana é uma métrica comum utilizada em diversos esquemas, inclusive em processamento de linguagem natural, como por exemplo em Mohammadhassanzadeh et al. (2020)

2.3.2 Distância de quarteirão

Seja $X_i = (x_{i1}, ..., x_{ip})^T$ vetor de observações da i-ésima amostra, $i = 1, ..., n$, no qual x_{ij} é o valor assumido pela variável x_j da amostra i. Artes e Barroso (2023) definem a distância de quarteirão (ou *Manhattan*) entre a i-ésima amostra e a k-ésima amostra como

$$d_{ik} = \sum_{j=1}^{p} |x_i - x_k|. \tag{2.3.3}$$

A Figura 2 ilustra geometricamente a distância de quarteirão entre dois pontos $\mathbf{X_i}$ e $\mathbf{X_k}$ no $\mathbb{R}^2$.

A distância de quarteirão é uma das alternativas à distância euclidiana, também com uso amplo na bibliografia, inclusive no contexto de processamento de linguagem

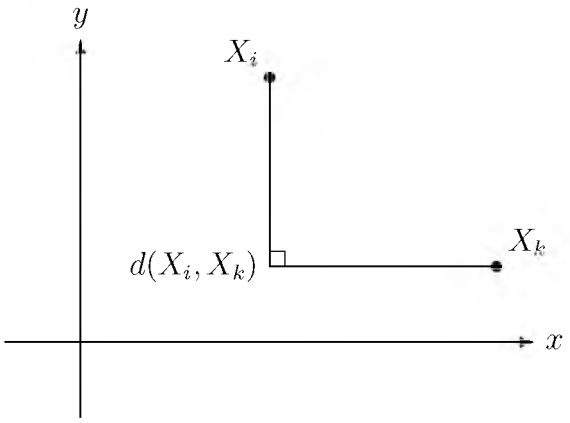

Figura 2: Distância de quarteirão entre os pontos $\mathbf{X}_i$ e $\mathbf{X}_k$.

Fonte: elaboração própria.

natural, como utilizado em Jotheeswaran e Kumaraswamy (2013).

2.3.3 Distância de Mahalanobis

Seja $\mathbf{X} = (x_1, ..., x_p)^T$ vetor aleatório com $\mathbb{E}(\mathbf{X}) = \mu = (\mu_1, ..., \mu_p)^T$, e $Cov(\mathbf{X}) = \mathbf{\Sigma}$. Artes e Barroso (2023) definem a distância de Mahalanobis ao quadrado entre $\mathbf{X}$ e μ como:

$$D^2_M(\mathbf{X}, \mu) = (\mathbf{X} - \mu)^T \mathbf{\Sigma}^{-1} (\mathbf{X} - \mu). \tag{2.3.4}$$

Seja $i = i, ..., n$ unidades amostrais de dimensão p. Logo, o vetor $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_n)$ define a amostra. Seja $\bar{\mathbf{x}}$ vetor de média desta amostra, e $\mathbf{S}$ matriz de covariância amostral. Logo, a distância de um ponto i em relação ao centróide da amostra será

$$D^2_M(\mathbf{x_i}, \bar{\mathbf{x}}) = (\mathbf{x_i} - \bar{\mathbf{x}})^T \mathbf{S}^{-1} (\mathbf{x_i} - \bar{\mathbf{x}}). \tag{2.3.5}$$

Pela definição trazida na Equação (2.3.4), necessitaríamos saber os valores μ e $\mathbf{\Sigma}$. Artes e Barroso (2023) argumentam que, em termos práticos, isto não é possível. Portanto, sugerem utilizar os estimadores usuais $\bar{\mathbf{x}}$ e $\mathbf{S}$, respectivamente, dando suporte para a definição 2.3.5. Os resultados passam a ser aproximados, com melhor desempenho para grandes amostras.

A distância de Mahalanobis é utilizada geralmente em problemas de detecção de *outliers* multivariados, como em Ghorbani (2019), sendo mais rara a utilização no contexto de processamento de linguagem natural.

2.3.4 Distância de cossenos

Seja $\mathbf{A}$ e $\mathbf{B}$ vetores não nulos. Então, pela Lei dos cossenos:

$$\mathbf{A} \cdot \mathbf{B} = |\mathbf{A}||\mathbf{B}| \cos\theta. \tag{2.3.6}$$

Ricardo e Berthier (2011) argumentam que podemos utilizar, para avaliar o grau de similaridade entre os vetores $\mathbf{A}$ e $\mathbf{B}$, a correlação entre eles. E para quantificar esta correlação, podemos utilizar o cosseno do ângulo entre estes vetores. Da Equação 2.3.6, temos que esta similaridade será:

$$\begin{aligned} \text{similaridade}(\mathbf{A}, \mathbf{B}) = \cos\theta &= \frac{\mathbf{A} \cdot \mathbf{B}}{|\mathbf{A}||\mathbf{B}|} \\ &= \frac{\sum_{i=1}^{n} \mathbf{A}_i \mathbf{B}_i}{\sqrt{\sum_{i=1}^{n} \mathbf{A}_i^2} \cdot \sqrt{\sum_{i=1}^{n} \mathbf{B}_i^2}}. \end{aligned}$$

A Figura 3, ilustra o ângulo θ formado entre os vetores $\mathbf{A}$ e $\mathbf{B}$.

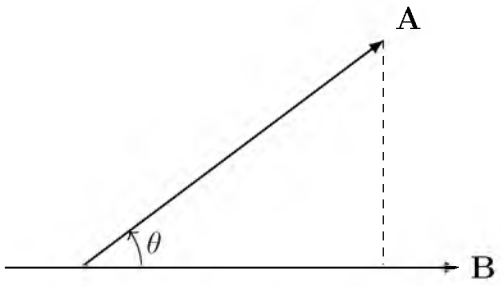

Figura 3: O cosseno como medida de similaridade entre $\mathbf{A}$ e $\mathbf{B}$

Fonte: elaboração própria.

Se considerarmos $\mathbf{A}$ e $\mathbf{B}$ os vetores numéricos (vetorização) das petições de dois processos, ou ainda da frequência de termos de um *Bag of Words*, conseguimos assim calcular uma medida de similaridade entre eles, calculando o complementar da medida de distância do cosseno.

Note que a distância do cosseno não respeita a desigualdade triangular definida em 2.3, portanto não pode ser considerada uma medida de distância estatística naquelas definições.

A distância do cosseno é possivelmente a métrica mais utilizada no paradigma do processamento de linguagem natural, sendo farta a bibliografia que utiliza desta medida

neste contexto, como por exemplo em Freitas (2023)

2.3.5 Divergência de Jensen-Shannon

Segundo Drost (2018), a divergência de Jensen-Shannon $JSD(P||Q)$ entre duas distribuições de probabilidade P e Q é definida por:

$$JSD(P||Q) = \frac{1}{2}(KL(P||R) + KL(Q||R)), \tag{2.3.7}$$

onde $R = \frac{1}{2}(P + Q)$ denota o ponto médio dos vetores de probabilidade P e Q, e $KL(P||R), KL(Q||R)$ denota a divergência de Kullback-Leibler de P e R, e Q e R.

As propriedades gerais da divergência de Jensen-Shannon são:

- $JSD(P||Q) \geq 0$;
- É uma medida simétrica, tal que $JSD(P||Q) = JSD(Q||P)$;
- $JSD(P||Q) = 0 \iff P = Q$.

Note que estas propriedades permitem utilizar a divergência de Jensen-Shannon como uma forma de medir a semelhança, definindo como $1 - JSD$ a "semelhança de Jensen-Shannon", que é o que de fato utilizo no escopo deste trabalho. Mais informações sobre esta medida podem ser obtidas em Drost (2018) e Burbea e Rao (1982). Mais detalhes sobre a divergência de Kullback-Leibler serão explicitados na seção 2.5.2

A divergência de Jensen-Shannon também é uma medida amplamente utilizada na bibliografia de processamento de linguagem natural, como em Lu, Henchion e Namee (2021)

2.3.6 Índice de Tversky

Tversky (1977) definiu este índice que serve como uma generalização de outros índices, como o índice de Jaccard, coeficiente de Sørensen–Dice e outros.

O índice é definido pela fórmula:

$$T(X, Y) = \frac{|X \cap Y|}{|X \cap Y| + \alpha|X \cap Y^C| + \beta|Y \cap X^C|}, \tag{2.3.8}$$

onde X e Y são conjuntos (de dados), $\alpha, \beta \geq 0$, $T(X, Y) \in (0, 1)$. Tomando X e Y como

vetores de frequências absolutas de termos em dois textos distintos e fixando valores α e β, é possível utilizar este índice para mensurar a semelhança entre esses textos.

2.3.7 Interseção Mínima de Probabilidade Empírica

Esta métrica foi desenvolvida a fim de ser uma modificação do índice de Tversky, tal que pudéssemos ter uma métrica inovadora que medisse de alguma forma a similaridade entre dois conjuntos de dados, mas ponderando de alguma forma a frequência relativa da ocorrência das palavras no corpo do texto, e não simplesmente a ocorrência ou não. Como a sensibilidade exigida pela métrica era alta, ela foi delineada para funcionar de forma conservadora, isto é, apenas acusando similaridade quando a similaridade entre as frequências relativas fosse bastante elevada.

Para cada conjunto de dados **A**, **B**, calcular a frequência relativa de cada termo **w**:

$$f_A(w) = \frac{\text{contagem}_A(w)}{\sum_{w' \in V} \text{contagem}_A(w')},$$

$$f_B(w) = \frac{\text{contagem}_B(w)}{\sum_{w' \in V} \text{contagem}_B(w')},$$

sendo V o conjunto de todos os termos presentes em ambos os textos e w' a soma da frequência de todos os termos.

Como estamos interessados em medir unicamente a força de interseção, somente serão considerados termos w que estão presentes em ambos os documentos.

Calculamos a diferença absoluta entre as frequências relativas dos termos em comum, tal que

$$dissimilaridade(w) = |f_A(w) - f_B(w)|.$$

Note que se a frequência relativa de um determinado termo w for igual em ambos os textos, a dissimilaridade será 0, ou seja, indicando que são iguais.

Por fim, tomaremos a soma das dissimilaridades, tal que

$$\text{IMPE}(\mathbf{A}, \mathbf{B}) = 1 - \sum_{w \in A \cap B} |f_A(w) - f_B(w)|, \tag{2.3.9}$$

definindo assim portanto a interseção mínima de probabilidade empírica (IMPE).

2.4 Métodos de agrupamento

Segundo Morettin e Singer (2023), o objetivo ao utilizar métodos de agrupamento é reunir dados pertencentes a um determinado espaço em grupos de acordo com alguma medida de distância, tal que minimize as distâncias dos elementos pertencentes a um mesmo grupo, e maximize as distâncias entre os grupos.

2.4.1 Vizinho mais próximo (KNN)

O método do vizinho mais próximo, também conhecido como *k-nearest neighbors* (KNN) pode ser definido como um método que utiliza as vizinhanças de uma observação a fim de classificar esta observação. Segundo Morettin e Singer (2023), o algoritmo associado à este método pode ser descrito por:

- Fixar $K > 0, K \in \mathbb{Z}$ e uma observação x_0;
- Identificar os K pontos do conjunto de dados que sejam mais próximos de x_0 segundo alguma medida de distância, e denotar este conjunto por ν_0;
- Estimar a probabilidade condicional de que o elemento selecionado do conjunto de dados pertença à classe j como a fração dos pontos de ν_0 cujos valores de Y sejam igual a j, ou seja, com

$$P(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in \nu_0} I(y_i = j); \tag{2.4.1}$$

- Classificar o elemento x_0 na j-ésima classe mais provável.

Na Figura 4, podemos observar o funcionamento do algoritmo KNN. Em um plano, observam-se 6 observações pertencentes a 2 classes distintas, além da observação x_0 a qual se deseja classificar em uma das duas classes. Caso o hiperparâmetro K seja definido como $K = 1$, a observação x_0 seria definida como pertencente a classe azul, pois o $K = 1$ ponto mais próximo pertence à classe azul. Caso o hiperparâmetro K seja definido como $K = 3$, a observação x_0 seria definida como pertencente a classe verde, pois a proporção das $K = 3$ observações que pertencem a classe verde é superior a quantidade de observações na classe azul. Numa situação em que sabemos as verdadeiras classes dos pontos, o hiperparâmetro K é passível de ser calibrado.

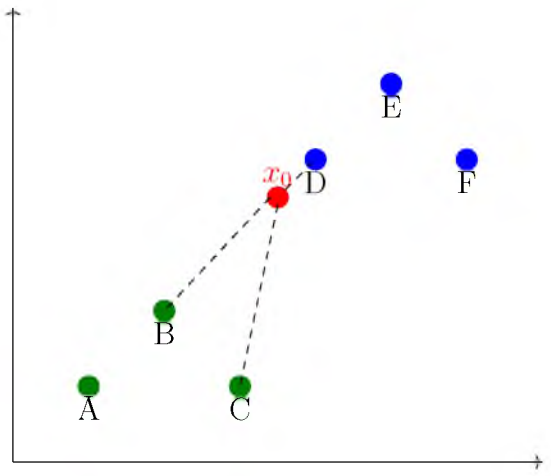

Figura 4: A observação x_0 sendo classificado segundo seus K vizinhos mais próximos

Fonte: elaboração própria.

O KNN não foi utilizado neste trabalho na sua forma tradicional, ou seja, de classificar pontos em j grupos de acordo com suas vizinhanças, mas a filosofia da técnica de buscar por similaridades de vizinhanças foi aplicada de forma adaptada ao escopo deste trabalho.

2.4.2 k-médias

k-médias, geralmente evocado como *k-means*, ou ainda k-medóides (Artes e Barroso, 2023) é um algoritmo que, segundo Morettin e Singer (2023), busca agregar pontos em K grupos de forma a minimizar a soma dos quadrados das distâncias euclidianas entre o centro (meio) de cada grupo e os pontos que o compõem.

Seja k o número de grupos em que se deseja agrupar os dados. Seja $\mathbf{x} = \mathbf{x}_1, ..., \mathbf{x}_n$ o vetor de amostra. O procedimento realizado pelo algoritmo pode ser descrito como:

- Fixar um número de k grupos, onde k é um hiperparâmetro do modelo;
- Alocar os elementos aleatoriamente aos k grupos, e calcular a medóide $\bar{\mathbf{x}}_k$ de cada grupo;
- Calcular a distância Euclidiana de cada item ao medóide do grupo em que está alocado:

$$d_{i,\bar{x}_k} = \sum_{k=1}^{k} \sum_{g_{(i)}=k} (\mathbf{x}_i - \bar{x}_k)^T (\mathbf{x}_i - \bar{x}_k), \tag{2.4.2}$$

em que $\bar{x}_k$ é a medóide do grupo k, e $g_{(i)}$ é o grupo que contém $\mathbf{x}_i$;

- Realocar cada item $\mathbf{x}_i$ ao grupo com medóide mais próximo tal que reduza $d_{i,\bar{x}_k}$. Então, calcular os novos valores de $\bar{x}_k$.
- Repetir os dois últimos passos de forma iterativa, até estabilizar o valor de $d_{i,\bar{x}_k}$.

No paradigma deste trabalho, o objetivo não é agrupar todos os processos em k grupos, mas encontrar processos semelhantes a um processo de referência. Ainda assim, o algoritmo k-médias pode ser útil para analisar de forma exploratória os dados, sendo possível testar diversos valores de k e analisar como os dados se agrupam em cada teste. Além disso, por se tratar de um algoritmo extremamente popular, torna possível a identificação e comparação com trabalhos análogos, e possibilitando o avanço do estado da arte do projeto com algoritmos mais sofisticados em uma possível futura aplicação. Para o que buscava-se responder com este trabalho, o algoritmo *k-means* não se mostrou conveniente, portanto não foi utilizado na aplicação final.

2.5 Visualização de dados

Uma parte crucial deste trabalho foi mostrar de forma simplificada as indicações de similaridades entre as petições iniciais fornecidas pela modelagem. Para isso, explorou-se algumas técnicas de visualização de dados para estes resultados, a fim de auxiliar no processo de tomada de decisões. Em geral, três ferramentas particularmente úteis para este tipo de modelagem são dendogramas, *t-SNE* e nuvens de palavras, que serão definidos a seguir.

2.5.1 Dendograma

Pela definição contida em Everitt e Skrondal (2010), dendograma é um diagrama geralmente utilizado para ilustrar a série de etapas executada por um método hierárquico, representando os passos de agrupamento do algoritmo. A altura do eixo y representa alguma medida de distância entre os agrupamentos. A Figura 5 ilustra um dendrograma genérico.

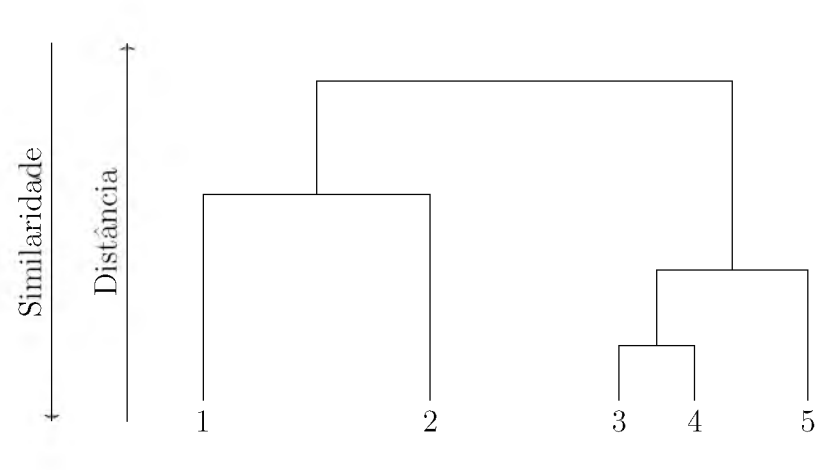

Figura 5: Exemplo simples de um dendograma

Fonte: elaboração própria.

2.5.2 t-SNE

Agrupamento estocástico de vizinhança com distribuição t-student — *t-distributed stochastic neighbor embedding (t-SNE)*, tradução livre — é uma técnica não linear de visualização de dados multidimensionais em duas (ou três) dimensões. Esta técnica busca apresentar a estrutura de todo um conjunto de dados multidimensional com escalas diversas em um só gráfico.

A Figura 6 traz o exemplo da técnica *t-SNE* para visualizar o conjunto de dados MNIST, de LeCun et al. (1998). Este conjunto de dados é constituído de algarismos escritos à mão por diversas pessoas diferentes, portanto com diversas caligrafias. Em geral, o objetivo da utilização deste conjunto de dados é para realizar agrupamento destes algarismos, com objetivo de verificar o quão bem uma técnica performa ao reconhecer os algarismos em diferentes caligrafias e sua eficácia em agrupá-los com seus semelhantes.

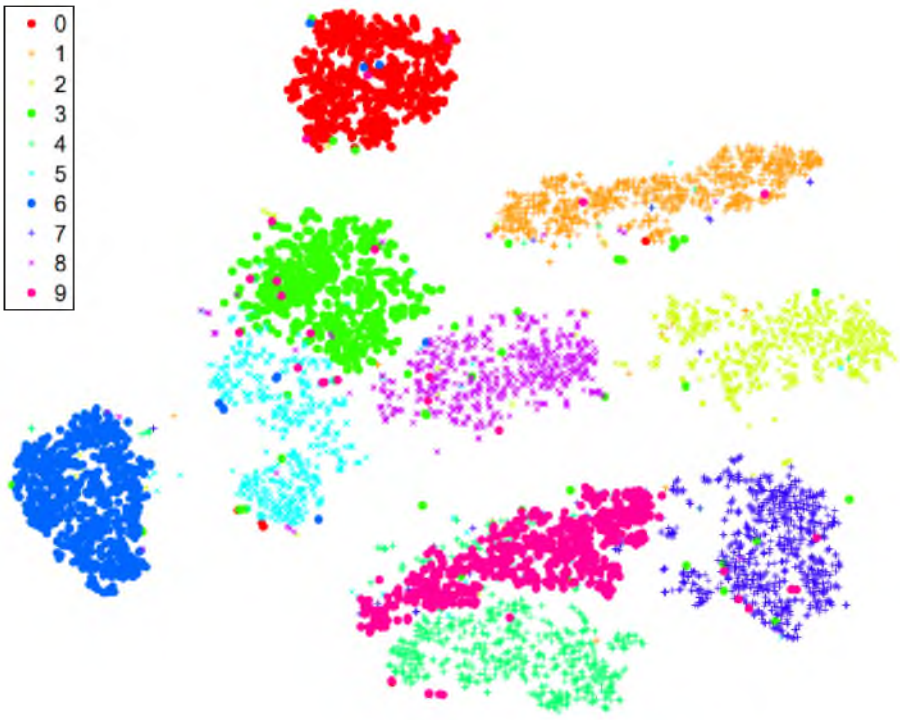

Figura 6: O algoritmo t-SNE aplicado aos dados MNIST.

Fonte: Maaten e Hinton(2008, p. 2590).

O conjunto de dados MNIST (LeCun et al, 1998) se tornou "padrão ouro" para teste de técnicas nas diversas áreas de reconhecimento de padrões, como análise multivariada clássica, processamento de linguagem natural, etc.

A técnica *t-SNE* busca apresentar de forma visual o resultado produzido pelo algoritmo utilizado. Por vezes, técnicas clássicas de visualização multivariada produzem visualizações de difícil interpretação, com interpolações dos agrupamentos formados na projeção dos resultados em duas dimensões. Para evitar estes problemas, utilizaremos a técnica *t-SNE*, em que Maaten e Hinton (2008) formularam o procedimento nos 4 passos a seguir:

- Passo 1: Converter distâncias Euclidianas de dados em probabilidades que representem similaridades

Na notação de Maaten e Hinton (2008), seja pontos x_j e x_i em um espaço N-dimensional, N grande. Iremos modelar a similaridade entre estes pontos como a probabilidade condicional $p(j|i)$ sendo a probabilidade condicional de x_j escolher x_i como seu vizinho, tal que estas probabilidades condicionais sejam relacionadas a similaridade entre

estes pontos. Para calcular esta probabilidade condicional de $p(j|i)$, Hinton e Roweis (2002) definem que esta probabilidade condicional $p(j|i)$ é proporcional a uma densidade de probabilidade gaussiana centrada em x_i e com variância σ_i $\forall i \neq j$, ou seja:

$$p(j|i) = \begin{cases} \frac{exp(-||x_i - x_j||^2/2\sigma_i^2)}{\sum_{k \neq i} exp(-||x_i - x_k||^2/2\sigma_i^2)}, \forall i \neq j; \\ 0, i = j. \end{cases} \tag{2.5.1}$$

Forçamos esta probabilidade $p(i|i) = 0$ pois estamos somente interessados em modelar similaridades pareadas.

- Passo 2: Mapear os pontos $x_1, ..., x_k$ em alta dimensionalidade para pontos $y_1, ..., y_k$ em dimensionalidade reduzida.

Seja $y_1, ..., y_k$ os pontos de dimensionalidade reduzida referentes aos pontos $x_1, ..., x_k$. Seja $q(j|i)$ a probabilidade condicional de y_j escolher o ponto y_i como seu vizinho, de acordo com a distância Euclidiana entre eles, e q_{ij} a probabilidade conjuta destes pontos. Segundo Maaten e Hinton (2008), a probabilidade conjunta q_{ij} é proporcional a uma densidade de probabilidade $t - student$ com 1 grau de liberdade (ou seja, proporcional à uma Cauchy), definindo assim:

$$q_{ij} = \begin{cases} \frac{(1+||y_i - y_j||^2)^{-1}}{\sum_{k \neq l}(1+||y_k - y_l||^2)^{-1}}, \forall i \neq j, \\ 0, i = j. \end{cases} \tag{2.5.2}$$

dessa forma, em contraposição assim ao algoritmo SNE proposto por Hinton e Roweis (2002), que utilizaria uma gaussiana ao invés da *t-student*. Esta troca por uma distribuição de cauda pesada é feita para evitar sobreposições de pontos, garantindo assim o padrão típico da visualização $t - SNE$, que faz uma separação mais robusta entre os agrupamentos, tornando a técnica extremamente eficaz em separar pontos dissemelhantes para visualização e interpretação gráfica. A justificativa teórica para utilizar a *t-student* é que "esta distribuição está diretamente relacionada à distribuição Gaussiana, dado que a distribuição *t-student* é uma mistura de infinitas Gaussianas"(Maaten e Hinton, 2008, p. 2586, tradução própria[3]), com médias iguais e precisões como variável aleatória gama.

- Passo 3: Minimizar a soma da divergência de Kullback-Leibler para as probabilidades conjuntas de todos os pontos, utilizando gradiente descentente.

[3]No original: "the Student t-distribution is that it is closely related to the Gaussian distribution, as the Student t-distribution is an infinite mixture of Gaussians".

A justificativa para o funcionamento da técnica, segundo Maaten e Hinton (2008), é que se o mapeamento dos pontos y_i, y_j — em dimensionalidade reduzida — modelar com fidedignidade a similaridade entre os pontos x_i e x_j, então as probabilidades condicionais $p(j|i)$ e $q(j|i)$ serão iguais. Portanto, o algoritmo busca justamente minimizar a diferença entre $p(j|i)$ e $q(j|i)$, e para isso, o algoritmo minimiza a soma da divergência de Kullback-Leibler (que neste caso é proporcional a entropia cruzada) para as probabilidades condicionais em todos os pontos utilizando o método de gradiente descendente. Por motivos de otimização computacional, o algoritmo busca minimizar a soma da divergência de Kullback-Leibler para as probabilidades conjuntas p_{ij}, definindo estas como simétricas tal que $p_{ij} = \frac{p(j|i)+p(i|j)}{2n}$, justificando assim a utilização da probabilidade conjunta em (2.5.2).

- Passo 4: Plotar os pontos y_k.

Note que, a termos práticos, $\mathbf{x_1}, ..., \mathbf{x_n}, \mathbf{y_1}, ..., \mathbf{y_n}$ serão vetores, de um conjunto $\mathbf{N}$ dimensional para os vetores $\mathbf{x_k}$, e $\mathbf{n} < \mathbf{N}$ dimensional para os vetores $\mathbf{y_k}$, sem perda de generalidade.

Portanto, o *t-SNE* busca separar no gráfico os agrupamentos formados, esclarecendo a estrutura dos resultados do procedimento de forma visualmente compreensível. Para mais detalhes do funcionamento da técnica, consultar Maaten e Hinton (2008).

Como observado, o objetivo do *t-SNE* é realizar uma redução de dimensionalidade fidedigna, a fim de mapear as instâncias — nesse trabalho, as petições inicias vetorizadas em vetores 300-dimensionais — em 2 dimensões para serem inseridas em um gráfico bidimensional.

2.5.3 Nuvens de palavras

Nuvens de palavras são uma forma simples de apresentar uma *Bag of Words* de forma visual e compreensível. Existem várias formas de apresentar uma nuvem de palavras, mas em geral iremos trazer os termos mais frequentes do documento, tal que o tamanho da palavra seja proporcional a frequência de ocorrência do termo no documento. Cores podem ser utilizadas tanto para compartimentalizar assuntos, quanto como simples elemento estético da figura.

Figura 7: Exemplo de nuvem de palavras

Fonte: elaboração própria.

A Figura 7 apresenta uma nuvem de palavras que foi construída com base nas palavras contidas nas petições iniciais do acervo de controle concentrado que foram utilizado como dado para este trabalho.

3 Metodologia

3.1 Conjunto de dados

O conjunto de dados deste trabalho é composto pelas petições iniciais de processos de controle concentrados tramitados ou em tramitação entre 1º de Janeiro de 2016 e 10 de Abril de 2024. Os dados são públicos e encontram-se disponíveis em Corte aberta. Optou-se pelo recorte ă partir de 2016 a fim de parear o recorte deste estudo com o recorte estabelecido por outras aplicações do Supremo Tribunal Federal, como a RAFA2030 (Freitas, Alencar e Rodrigues, 2022).

Os documentos originais (petições iniciais) são recebidos pelo STF em formato PDF. Entretanto, os textos ocerizados e pré-processados foram fornecidos pela Secretaria de Tecnologia da Informação (STI) do Tribunal para a manufatura deste trabalho.

Como a seleção dos processos mais semelhantes depende dos dados que compõem o acervo dos processos em tramitação do Tribunal em uma determinada data, testou-se as técnicas escolhidas em diversos recortes temporais dos dados obtidos, considerando em cada teste apenas os dados que compunham o acervo na data fixada de cada teste.

Não existe uma forma analítica simples de avaliar quais processos compunham o acervo em uma determinada data. De forma simplificada, um processo compõe o acervo no início de sua autuação, e deixa de compor quando tem baixa definitiva. Entretanto, retiradas de pauta, baixas temporárias, suspensões, trocas de relatoria e outras situações eventuais podem mover o processo como pertencente ou não ao acervo. Apesar disso, essas exceções serão desconsideradas para a determinação da composição das amostras, visto que simplesmente ao considerar a diferença da data da baixa pela data de autuação do processo, é possível determinar com grande fidedignidade o acervo na data selecionada (especialistas do tribunal afirmam que é possível obter mais de 95% da composição do acervo desta forma). Como a intenção da aplicação é ser utilizada no acervo atual — do qual sabemos com precisão o acervo da data presente em tempo real —, sendo as datas passadas apenas para avaliação do modelo, esta métrica será utilizada sem embargos para definição da composição de cada amostra.

Visto que o Painel de Transparência do STF traz um quantitativo de acervo de controle concentrado por ano, optou-se por realizar um teste para cada um dos anos do recorte temporal estabelecido (2016 a 2024), fixando a data em primeiro de janeiro de cada ano. Portanto, foram nove conjuntos de amostra em que a aplicação foi testada para

avaliação de sua eficácia, e esta foi finalmente implementada com a data final disponível nos dados (10 de Abril de 2024), simulando o presente.

Amostra	Data	Dados disponíveis	Acervo real	Porcentagem (%)
1	01/01/2016	1675	2217	75
2	01/01/2017	1807	2311	78
3	01/01/2018	1913	2240	85
4	01/01/2019	1879	2071	90
5	01/01/2020	1766	1784	99
6	01/01/2021	1521	1739	87
7	01/01/2022	1489	1511	98
8	01/01/2023	1240	1240	100
9	01/01/2024	1049	1118	94

Tabela 1: Integridade das Amostras

Fonte: elaboração própria.

A Tabela 1 apresenta a integridade dos dados disponíveis para cada partição amostral, portanto variando de 75% à 100% do acervo real em cada uma das datas escolhidas.

Esse conjunto de dados foi complementado com algumas covariáveis relacionadas aos respectivos processos utilizando as informações disponíveis no Painel de Transparência do STF. Entretanto, essas covariáveis não foram utilizadas em nenhuma etapa da modelagem, e só foram inseridas ao final para serem analisadas pelo usuário final da aplicação.

Todos os conjuntos de dados se mostraram extremamente pesados e consumidores de memória, trazendo a necessidade da utilização de tabelas virtuais para viabilizar os tempos de processamento de cada uma das técnicas. Portanto, utilizou-se o pacote *RSQLite* para construção de uma base de dados, onde foram inscritas todas as tabelas pesadas utilizadas neste trabalho, isto é, dados crus, dados processados, resultados parciais, amostras, resultados de cálculos de métricas e demais tabelas pesadas para que fossem acessadas no decorrer do trabalho até a aplicação via linguagem *SQL* pelo próprio R utilizando o pacote supracitado.

3.2 Métodos

As metodologias utilizadas nesse trabalho nas etapas de pré-processamento dos dados, vetorização dos textos e busca de similaridades estão descritas nessa seção.

Nesse trabalho foram utilizados diversos procedimentos de limpeza de palavras

vazias. Mais especificamente, no modelo *doc2vec* pré-treinado do tribunal, não se sabe ao certo quais as palavras foram consideradas palavras de parada e portanto filtradas nos seus dados de treinamento, pois detalhes dos parâmetros deste modelo não podem ser revelados por questões de segurança de informação. Já para a comparação de similaridade dos corpos textuais, não foi aplicada nenhuma rotina de limpeza de palavras vazias, com excelentes resultados. Por fim, para a construção de nuvens de palavras para compreensão da natureza textual dos textos de forma simples e rápida pelo usuário, construiu-se uma extensa lista de filtragem de palavras, muitas das quais não são por vezes consideradas palavras vazias, mas que para o objetivo específico deste procedimento, não contribuíam com a compreensão da natureza do processo (por exemplo, filtrou-se datas, endereços, nomes e palavras comuns no universo do direito como advogado, juiz, tribunal e demais palavras que não contribuíam diretamente com a compreensão do quão parecido dois processos são entre si). Estas rotinas escolhidas possibilitam que esta aplicação seja utilizada como paradigma para um posterior avanço no estado da arte da aplicação.

Para a vetorização dos textos, uma das técnicas utilizadas neste trabalho foi o método *doc2vec*, retornando vetores numéricos de dimensão 300. Com isso, além de buscar garantir conformidade com demais tecnologias do tribunal que também utilizam alguma rotina de NLP, certificamos que o contexto semântico gerado pela vetorização estivesse em conformidade com o vocabulário jurídico que se espera nos dados, visto que utilizou-se um modelo *doc2vec* pré-treinado da STI do tribunal. A utilização de um modelo pré treinado com dados do tribunal busca assegurar o significado léxico dos vetores formados, uma vez que um modelo treinado com dados do escopo jurídico garante uma menor quantidade de ruído quanto às expressões encontradas nos dados. Por exemplo, este modelo quase certamente não irá confundir a palavra Juiz com um contexto futebolístico de um árbitro de futebol, entendendo portanto que se refere a um magistrado no contexto da frase. Assim, evita-se a necessidade de treinamento do modelo com quantidades massivas de dados como por vezes se mostra necessário em modelos de linguagem de grande escala para usos gerais, tornando viável a utilização desta aplicação em computadores pessoais com capacidades computacionais limitadas em um tempo significativamente baixo.

Para utilizar o modelo *doc2vec* pré-treinado da STI do tribunal, foi criado um ambiente virtual *Conda* a fim de tornar os códigos indefectíveis, por meio do pacote *gensim*, além de utilização do módulo *polars* — ainda que muito menos popular que o *pandas* para manipulação de dados, testes recentes indicam ser muito mais efetivo e rápido (ver Nahrstedt et al., 2024), o que justifica a escolha. Como fazia parte do paradigma do trabalho a preferência pela linguagem *R*, utilizou-se esta etapa no *Python* de forma

minimalista com uma chamada via o pacote *reticulate*, realizando apenas o estritamente necessário em *Python* e seguindo com a aplicação das demais técnicas em *R*.

Para empregar a técnica de vetorização *Bag of Words*, foi desenvolvida uma função no R que utiliza uma séria de funções implementadas no pacote *tm*, tal que a entrada da função seja o texto do processo (a petição inicial), e a saída da função seja uma tabela com duas colunas e n linhas, sendo a primeira coluna o termo, e a segunda coluna a frequência absoluta da aparição do termo na petição inicial. A quantidade n de linhas é variável para cada documento, sendo igual a quantidade de termos únicos do documento.

Preocupações éticas ao aplicar técnicas de vetorização devem ser sempre observadas. Estereótipos e fatores de gênero nos dados podem ser criados, mantidos ou até amplificados como observado por Bolukbasi et al. (2016). Aplicações ingênuas desta técnica podem servir para perpetuação de vieses sociais deletérios sob a justificativa de serem produzidos por uma máquina supostamente imparcial, como o "Racismo algorítmico"descrito por Silva (2022). Esta aplicação propôs-se tão somente a auxiliar na identificação de similaridades de processos em sua fase inicial de tramitação, sem qualquer interesse em substituir uma posterior análise humana rigorosa de seu escopo — e muito menos de substituir um magistrado quanto ao mérito do conteúdo do processo — evitando assim que esta aplicação seja uma responsável oculta na perpetuação destes vieses.

A partir da obtenção dos textos e seus respectivos vetores numéricos, realizou-se a construção das amostras, respeitando os tempos de autuação e baixa dos processos. Portanto, construiu-se estas amostras representando o acervo de processos de controle concentrado do tribunal no dia 1º de Janeiro dos anos de 2016 à 2024, além do acervo representando a data atual sintética, isto é, a última data presente na amostra (10 de Abril de 2024).

Para cada uma dessas amostras, definiu-se um processo de referência (ou paradigma) e calculou-se medidas rotineiras de similaridade aplicadas em processamento de linguagem natural, como distância do cosseno e distância euclidiana entre os processos e a distância da visualização em duas dimensões proporcionada pelo t-SNE, a fim de identificar os processos mais semelhantes ao processo paradigma. Ou seja, foi aplicada a metodologia semelhante ao algoritmo KNN usando diversas medidas de distâncias, buscando comparar a eficácia de cada um dos métodos.

Um dos desafios deste trabalho foi a não existência de dados observados para validar a escolha dos processos mais parecidos ao processo de referência. Para obter

alguma forma de validação das seleções propostas pelos diferentes métodos, observou-se que, no histórico de processos pautados no plenário, existem processos de controle concentrado que de alguma forma tiveram pauta e/ou decisão conjunta, visto o tema ser semelhante, como por exemplo em Informação à sociedade — ADIs 6.040 e 6.055. Observando este recurso, inferiu-se que, independente do motivo e forma que o Tribunal entendeu estes processos como similares, dever-se-ia obter de forma automatizada um resultado análogo da similaridade destes processos. Portanto, selecionou-se aleatoriamente alguns dos processos disponíveis publicamente em que existiam processos de controle concentrado pautados conjuntamente, e buscou-se por meio das técnicas e métricas deste estudo calcular os processos mais similares e comparar com os indicados pelo tribunal, segundo esta lógica, como similares.

Para todas as técnicas de vetorização utilizadas neste trabalho referentes as petições iniciais dos processos, estudou-se a utilização de medidas de similaridade e dissimilaridade para comparação desses processos.

4 Resultados

Este trabalho se propôs a identificar processos de possível pauta ou encaminhamento conjunto utilizando técnicas de processamento de linguagem natural. Aplicada sobre a primeira fonte de informação que chega ao tribunal referente a um determinado processo — sua petição inicial —, fornece uma ferramenta aos analistas do STF para agilizar algumas etapas de encaminhamento processual no Tribunal. Para isso, foram testadas técnicas de vetorização de textos e métricas de similaridade amplamente utilizadas pela bibliografia, bem como a proposta de uma nova métrica — sendo essa a modificação de uma já existente.

4.1 Vetorização dos textos

Para analisar a similaridade dos textos, utilizou-se três técnicas de vetorização: *doc2vec*, *Bag of Words* completa e *Bag of Words* reduzida. A primeira técnica foi a transformação dos textos das petições iniciais em vetores numéricos utilizando o modelo *doc2vec* pré-treinado do Tribunal. A segunda técnica foi a construção de *Bag of Words* completo, que foi realizada da seguinte forma:

- Regularização das palavras removendo pontuações, transformando letras maiúsculas em minúsculas, e remoção dos termos entendidos enquanto palavras vazias como artigos, numerais e conectivos;
- Calculo da frequência absoluta e relativa de cada termo no texto.

A terceira técnica foi a construção de *Bag of Words* reduzido, que foi realizada da seguinte forma:

- Regularização das palavras removendo pontuações, transformando letras maiúsculas em minúsculas, e remoção dos termos entendidos enquanto palavras vazias como artigos, numerais e conectivos;
- Calculo da frequência absoluta e relativa de cada termo no texto;
- Selecionar apenas os 100 termos mais frequentes.

4.2 Estudos de caso

As medidas de distância (similaridades) escolhidas para comparação dos textos foram: distância do cosseno, euclidiana, divergência de Jensen-Shannon e convergência mínima de probabilidade empírica (IMPE). Para os estudos de caso, foram selecionados alguns processos de forma conveniente, tendo em vista as informações de processos que foram pautados juntos no plenário do Tribunal. Para cada processo, as três formas de vetorização descritas na seção 4.1 foram aplicadas. Por fim, comparou-se qual combinação de métrica e vetorização melhor performou em selecionar os vizinhos mais próximos do processo paradigma.

4.2.1 Estudo de caso 1

Processo paradigma: ADI6931.

Processo analisado conjuntamente: ADI6921.

Vetorização	**Métrica de similaridade**	**Proximidade**
doc2vec	Distância do cosseno	8º mais similar
Bag of Words completo	Distância do cosseno	**1º mais similar**
	IMPE	65º mais similar
	Jensen-Shannon	**3º mais similar**
Bag of Words reduzido	Distância do cosseno	7º mais similar
	IMPE	**4º mais similar**
	Jensen-Shannon	72º mais similar

Tabela 2: Similaridades entre ADI6931 e ADI6921.

Top 10 palavras ADI6931	**Top 10 palavras ADI6921**
distribuição, assinatura, consultoria, **serviço**, carregamento, **telecomunicações**, canais, distribuidoras, obrigatório, conteúdo	**telecomunicações**, emenda, devido, serviços, página, **serviço**, impugnado, programação, NA, NA

Tabela 3: Tabela de palavras mais frequentes

Na Tabela 3, é possível obter uma breve compreensão do conteúdo dos processos

ADI 6931 e ADI 6921. Na primeira coluna, lista-se as dez palavras mais frequentes do processo ADI 6921, enquanto na segunda coluna, observa-se as palavras mais frequentes da ADI 6931. Em negrito, destacam-se os termos em comum em ambos os documentos dentre as dez palavras mais frequentes de cada um.

4.2.2 Estudo de caso 2

Processo paradigma: ADPF857.

Processos analisados conjuntamente: ADPF743 e ADPF746.

Vetorização	**Métrica de similaridade**	**Proximidade**
doc2vec	Distância do cosseno	**1º e 2º mais similar**
Bag of Words completo	Distância do cosseno	**1º e 2º mais similar**
	IMPE	42º e 89º mais similar
	Jensen-Shannon	**1º e 3º mais similar**
Bag of Words reduzido	Distância do cosseno	**1º e 2º mais similar**
	IMPE	8º e 9º mais similar
	Jensen-Shannon	**1º e 2º mais similar**

Tabela 4: Similaridades entre ADPF857 e ADPF743, ADPF746.

Top 10 palavras ADPF857	**Top 10 palavras ADPF746**	**Top 10 palavras ADPF743**
incêndios, **pantanal**, **partido**, prevenção, **diretório**, sustentabilidade, **trabalhadores**, combate, **ambiente**, maio	**ambiente**, queimadas, **pantanal**, **partido**, **trabalhadores**, ambiental, **diretório**, governo, fogo, amazônia	ambiental, **ambiente**, rede, vida, **incêndios**, **pantanal**, desmatamento, idade, amazônia, saúde

Tabela 5: Tabela de palavras mais frequentes

Na Tabela 5, observa-se as palavras mais frequentes nos processos ADPF 857, ADPF 746 e ADPF 743. Em negrito, as interseções entre as palavras mais frequentes do processo ADPF 857 com os demais.

4.2.3 Estudo de caso 3

Processo paradigma: ADO54.

Processo analisado conjuntamente: ADPF760.

Vetorização	**Métrica de similaridade**	**Proximidade**
doc2vec	Distância do cosseno	**4º mais similar**
Bag of Words completo	Distância do cosseno	**1º mais similar**
	IMPE	36º mais similar
	Jensen-Shannon	**3º mais similar**
Bag of Words reduzido	Distância do cosseno	**1º mais similar**
	IMPE	8º mais similar
	Jensen-Shannon	**2º mais similar**

Tabela 6: Similaridades entre ADO54 e ADPF760.

Top 10 palavras ADO54	**Top 10 palavras ADPF760**
desmatamento, **disponível**, **amazônia**, **ambiente**, rede, **ambiental**, dados, economia, abilidade, aumento	**desmatamento**, **amazônia**, **ambiental**, **ambiente**, **disponível**, povos, indígenas, procdam, mudanças, climáticas

Tabela 7: Tabela de palavras mais frequentes

Observando a Tabela 7, pelos termos mais frequentes contidos nos processos ADO 54 e ADPF 760, nota-se que o modelo foi sensível em observar diferenças importantes ante aos processos observados na Tabela 5, ainda que o conteúdo seja parecido. A similaridade entre os processos selecionados em 4.2.2 Estudo de Caso 2 e 4.2.3 Estudo de caso 3, não foi reconhecida pelo Tribunal e nem por este modelo, indicando que o nível de acurácia observado é satisfatório e em harmonia com o entendimento do Tribunal.

4.2.4 Estudo de caso 4

Processo paradigma: ADI7078.

Processos analisados conjuntamente: ADI7066 e ADI7070.

Vetorização	**Métrica de similaridade**	**Proximidade**
doc2vec	Distância do cosseno	1º e 6º mais similar
Bag of Words completo	Distância do cosseno	**1º e 2º mais similar**
	IMPE	437º e 475º mais similar
	Jensen-Shannon	**1º e 2º mais similar**
Bag of Words reduzido	Distância do cosseno	**1º e 2º mais similar**
	IMPE	**3º e 4º mais similar**
	Jensen-Shannon	**1º e 2º mais similar**

Tabela 8: Similaridades entre ADI7078 e ADI7066, ADI7070.

Top 10 palavras ADI7078	**Top 10 palavras ADI7066**	**Top 10 palavras ADI7070**
ceará, **icms**, **tributária**, fortaleza, alencar, bárbara, centro, fone, martins, queiroz	**icms**, atra, operações, abimao, imposto, cobrança, contribuinte, difal, **tributária**, anterioridade	**icms**, alagoas, operações, **tributária**, difal, governador, gabinete, anterioridade, contribuinte, alíquota

Tabela 9: Tabela de palavras mais frequentes

Da Tabela 9, nota-se que a presença de *stopwords* não foi um impeditivo para que o modelo utilizando a vetorização *Bag of Words* completa, combinada com a métrica da distância do cosseno identificasse com sucesso os dois processos entendidos como similares pelo Tribunal.

4.2.5 Estudo de caso 5

Processo paradigma: ADI3318.

Processos analisados conjuntamente: ADI2943 e ADI3309.

Vetorização	Métrica de similaridade	Proximidade
doc2vec	Distância do cosseno	**1º e 5º mais similar**
Bag of Words completo	Distância do cosseno	**3º e 4º mais similar**
	IMPE	358º e 360º mais similar
	Jensen-Shannon	**2º e 3º mais similar**
Bag of Words reduzido	Distância do cosseno	**2º e 3º mais similar**
	IMPE	6º e 7º mais similar
	Jensen-Shannon	**2º e 4º mais similar**

Tabela 10: Similaridades entre ADI3318 e ADI2943, ADI3309.

Top 10 palavras ADI3318	Top 10 palavras ADI2943	Top 10 palavras ADI3309
criminal, **policial**, **penal**, **investigação**, **inquérito**, **polícia**, **judiciária**, **procedimento**, **investigações**, externo	**inquérito**, **penal**, **policial**, **polícia**, **criminal**, **investigação**, **procedimento**, **judiciária**, administrativo, habeas	**polícia**, **penal**, **inquérito**, **investigação**, **policial**, **investigações**, **criminal**, **judiciária**, criminais, **procedimento**

Tabela 11: Tabela de palavras mais frequentes

Pela Tabela 11, pode-se presumir que estamos tratando de processos criminais. Neste caso, observa-se que existem diversos processos criminais no acervo, e o conteúdo textual deles é bastante similar — o que pode explicar o motivo pelo qual nenhuma das vetorizações e métricas de similaridade conseguiram indicar como 1º e 2º processos mais similares as ADI 2943 e ADI 3309 na Tabela 10. Ainda assim, nota-se que várias métricas foram bem sucedidas em identificar estes processos como razoavelmente similares, inclusive a vetorização *Bag of Words* combinada com a distância do cosseno. Isto pode indicar que esta técnica pode não ser tão sensível para processos de natureza criminal, portanto sendo necessário, neste caso, analisar alguns processos a mais dentre os indicados como mais similares.

4.2.6 Análise

Os estudos de caso mostram que as metodologias testadas foram eficientes em encontrar, para a maioria dos casos, processos que de fato contém similaridades pertinentes para o meio jurídico, utilizando como validador desta hipótese processos em recorte do acervo que tiveram pauta conjunta pelo plenário da Corte.

Notou-se, portanto, que a utilização do *Bag of Words* completo mostrou-se a forma mais eficaz de vetorização neste contexto, combinada com a métrica de similaridade da distância do cosseno — sendo este ligeiramente melhor em alguns casos que os *doc2vec*, e significativamente melhor que os *Bag of Words* reduzido. Os resultados indicam que também poderia ser utilizado os vetores numéricos (*tokens*) produzidos pelo *doc2vec*, porém como este requer etapas computacionalmente intensivas e complexas para obtenção de resultados iguais ou até ligeiramente piores, determinou-se, pelo princípio da parcimônia, não utilizar esta vetorização, e seguir pela via mais simples, isto é, a utilização de *Bag of Words*. Ou seja, dispensa-se nesse caso a necessidade de um modelo elaborado para o *embedding* do texto, como o *doc2vec*.

Além disso, observa-se que a melhor medida de distância foi a distância do cosseno, seguida pela semelhança de Jensen-Shannon. Não obstante, a distância do cosseno é a medida mais utilizada no contexto de NLP. Desta forma, para a aplicação final, o método adotado para seleção dos processos mais similares é selecionar os processos com menor valor de distância do cosseno — na forma do seu complementar, transformando assim em maior valor para similaridade do cosseno — aplicada sob a *Bag of Words*.

4.2.7 t-SNE

Realizou-se também estudos de casos envolvendo a construção de gráficos utilizando o *t-SNE* para analisar se a visualização fornecida por este método trazia com fidedignidade as semelhanças e dissemelhanças buscadas. Para a construção destes gráficos, utilizou-se os *tokens* numéricos obtidos da vetorização *doc2vec* como os dados numéricos para entrada no algoritmo, a fim de observar se as visualizações fornecidas seriam compatíveis com os resultados encontrados no estudo de métricas, fornecendo assim mais um insumo visual para a identificação dos processos similares no acervo. Para comparar a efetividade deste mapeamento, fixou-se um processo paradigma, e calculou-se os 10 processos mais próximos segundo a distância do cosseno, a distância euclidiana e a proximidade sugerida pelo algoritmo do *t-SNE*. Os resultados estão ilustrados nas Figuras 8,

9 e 10.

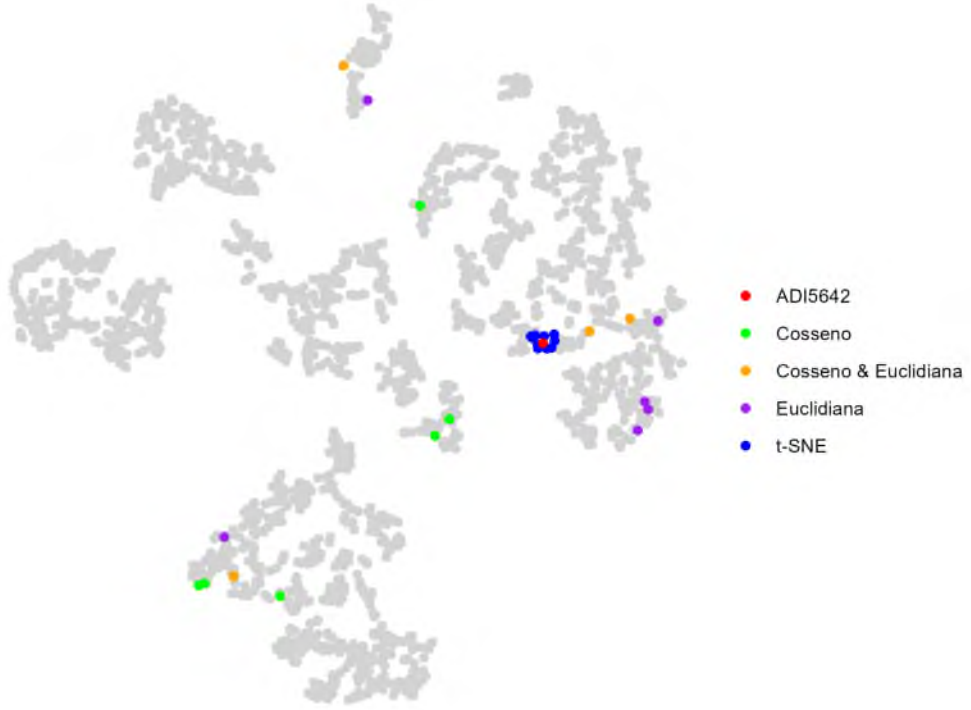

Figura 8: t-SNE do processo ADI5642 no acervo.

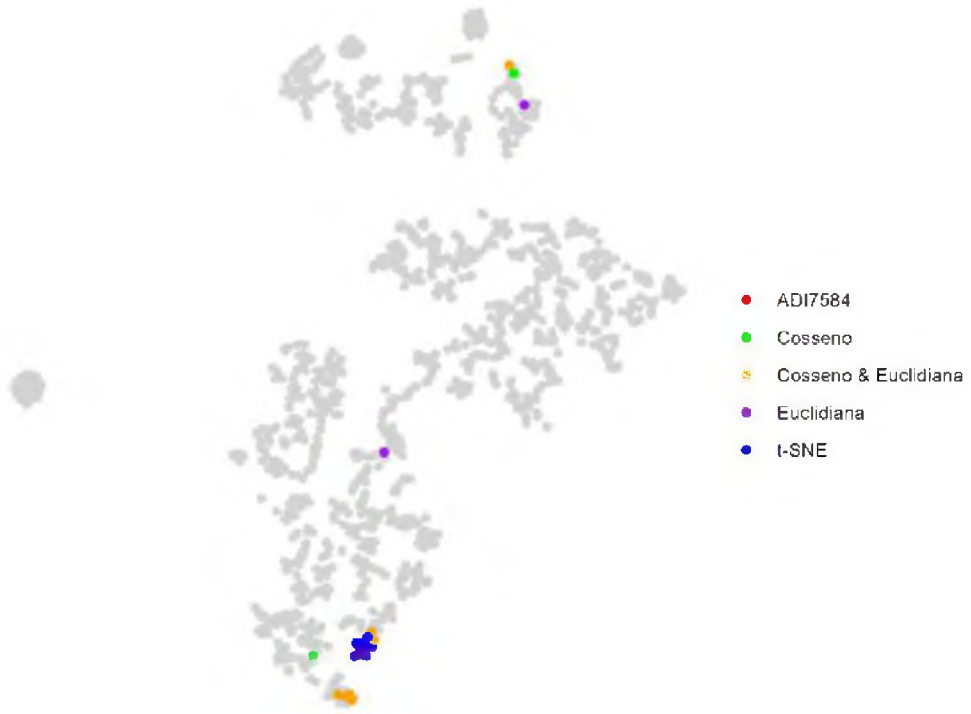

Figura 9: t-SNE do processo ADI7584 no acervo.

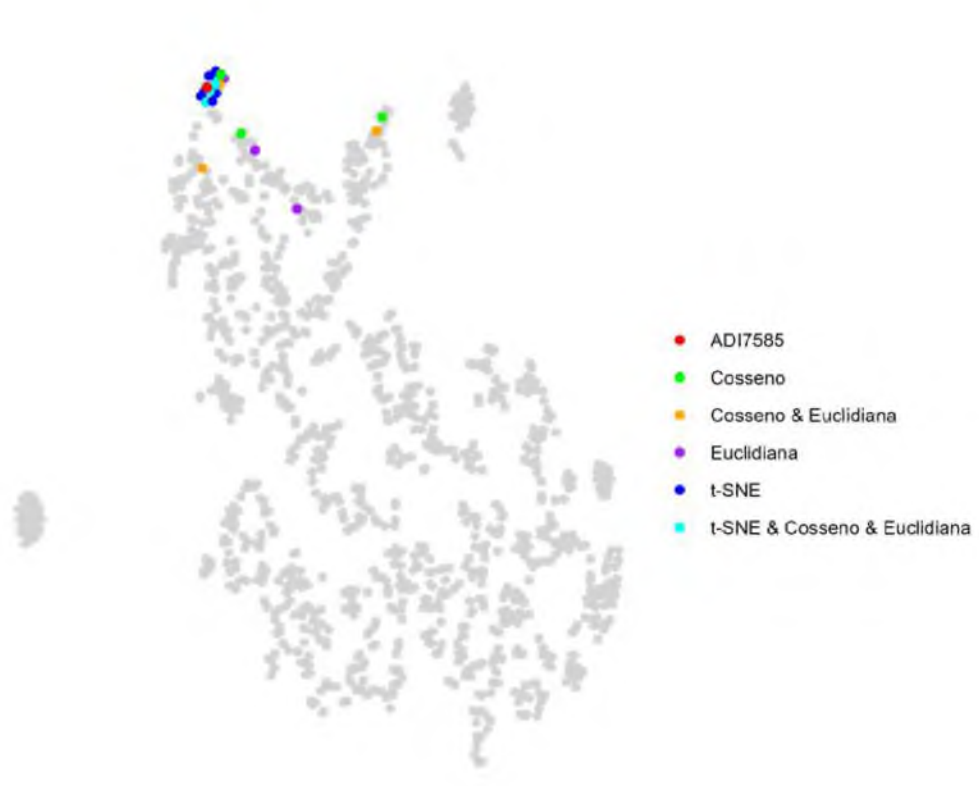

Figura 10: t-SNE do processo ADI7585 no acervo.

Nota-se nas Figuras 8, 9 e 10, que os processos identificados como mais próximos de um processo fixado pela técnica t-SNE não coincidem com os identificados como mais próximos segundo métricas de distância amplamente aceitas e utilizadas na bibliografia, como a distância do cosseno e a distância euclidiana. Na realidade, a técnica em geral afastou os processos que deveriam ser identificados como similares, de forma que em raros casos formou agrupamentos coerentes, e em geral desfiguraram as proximidades esperadas dos processos.

Para a rotina de vetorização *doc2vec* desta aplicação, utilizou-se como dimensão dos vetores de características produzidos pelo modelo sob os dados inseridos o tamanho 300, padrão definido pelo tribunal. O mapeamento produzido pelo *t-sne* reduz esta dimensão de tamanho 300 para 2 dimensões, a fim de se obter coordenadas cartesianas para visualização gráfica em duas dimensões. Esta redução significativa de dimensionalidade não foi capaz de preservar as estruturas de similaridade esperadas dos processos, produzindo resultados insatisfatórios como observado nas Figuras 8 e 9.

4.3 Aplicação

A fim de efetivamente utilizar os resultados obtidos neste trabalho, foi construído um aplicativo *Shiny* no *software* R. Inicialmente, o aplicativo estará disponível somente para público interno do Tribunal, podendo ser publicado para o público externo em momento oportuno.

Considerando que o método que retornou os melhores resultados foi a vetorização *Bag of Words* completo, com a métrica de similaridade da distância do cosseno, essa foi a configuração implementada no aplicativo *Shiny*. Além disso, para visualização dos temas dos processos selecionados, o aplicativo traz nuvens de palavras dos mesmos para comparação visual.

Pela utilização de metodologias computacionalmente eficientes, esta aplicação se mostra tanto útil quanto prática, permitindo ao usuário um alto nível de interação com a aplicação. Conforme ilustrado na Figura 11, o usuário inicialmente informa o processo paradigma e o número de processos similares a ele que se deseja encontrar. O tamanho da nuvem de palavras desejada também é outra variável disponível.

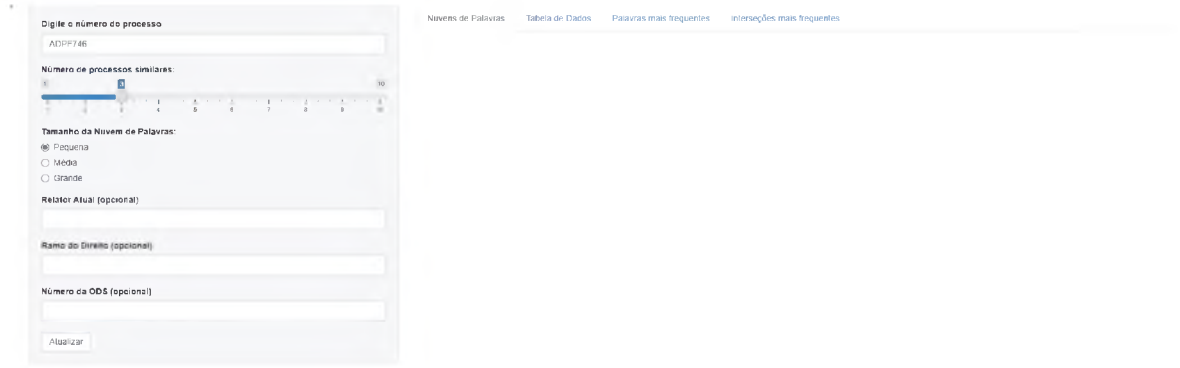

Figura 11: A aplicação ao ser executada.

A Figura 12 apresenta um exemplo da navegação pelos filtros da aplicação *Shiny*.

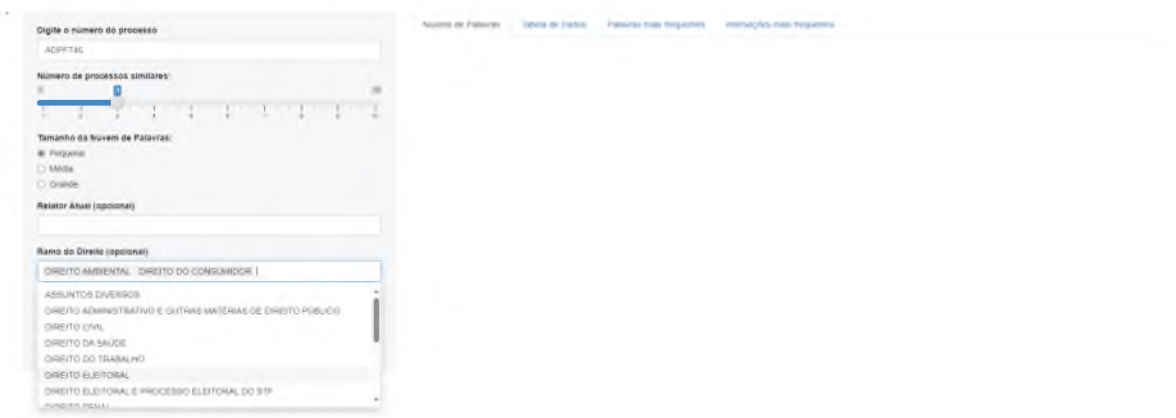

Figura 12: Navegação pelos filtros da aplicação.

Como observado na Figura 12, é delegado ao usuário a possibilidade de restringir a busca utilizando covariáveis relevantes no contexto jurídico. Os filtros foram construídos de forma fechada, permitindo ao usuário selecionar com o *mouse* as categorias que deseja trazer na aplicação, evitando assim possíveis erros de digitação do usuário. Caso o usuário não selecione nenhum filtro para as categorias, então a aplicação retornará todos os resultados possíveis.

Definidos os filtros, a aplicação irá retornar os processos calculados como mais similares pela distância do cosseno aplicada sobre a *Bag of Words* do texto completo da petição inicial, conforme ilustrado na Figura 13. A nuvem de palavras do processo paradigma será automaticamente gerada.

Figura 13: Retorno da aplicação após selecionar filtros e clicar em Atualizar.

Para auxiliar o usuário a obter uma noção geral do conteúdo do processo, foram construídas nuvens de palavras utilizado a *Bag of Words*. Porém, foi adicionada uma etapa a mais de filtragem de palavras vazias bem mais agressivos. Foram removidas quase duas mil palavras frequentes no universo jurídico, que são comuns a quase todos os processos, e não seriam convenientes para uma rápida comparação entre processos, como por exemplo: nomes de juízes, ministros, advogados, siglas comuns de endereços, palavras como "constituição", "justiça", "tribunal", etc. Com isso, buscou-se deixar apenas palavras-chave que caracterizam um processo, como tributário, indígena, trabalhista, etc. É pretendido que esta lista criada abranja um extenso leque de possibilidades de palavras vazias, porém caso esta lista se mostre insuficiente com o tempo, ela pode ser facilmente estendida, complementando com demais palavras que deveriam ser filtradas visto que foi implementada como um arquivo separado em formato *txt* que pode ser facilmente alterado pelo usuário ou pelos responsáveis pela manutenção desta aplicação.

Figura 14: Nuvens de palavras após clicar em caixas de seleção.

Conforme ilustrado na Figura 14, o usuário poderá selecionar até 3 processos para ilustração das nuvens de palavras, em conjunto com a nuvem do processo paradigma, permitindo uma primeira comparação visual do conteúdo dos processos selecionados. Neste exemplo, nota-se que os processos versam sobre temas ambientais, incêndios e seus impactos.

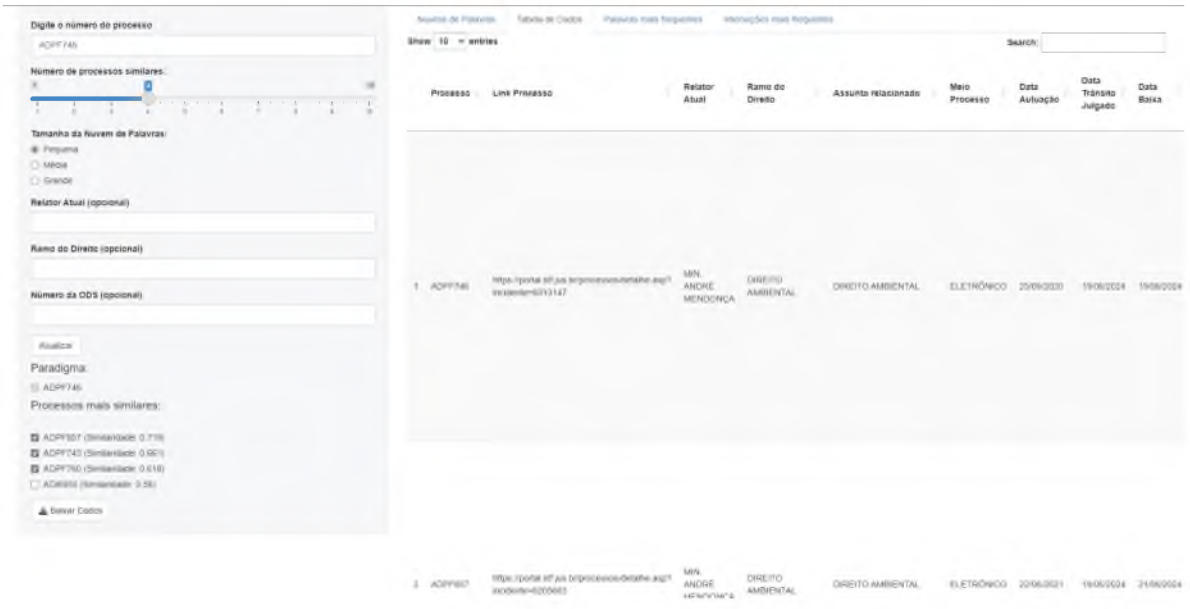

Figura 15: Segunda aba da aplicação — lista de dados.

Na segunda aba da aplicação, Figura 15, é apresentada a lista de covariáveis dos processos selecionados e do processo paradigma, este na primeira linha. Em uma

situação que o processo paradigma ainda não tenha covariáveis — por exemplo, caso tenha acabado de chegar ao tribunal e não tenha nada senão a sua petição inicial preenchida — a aplicação irá funcionar normalmente, o processo paradigma nesta situação viria com todas as covariáveis em branco, à exceção de sua classe e número (que são os primeiros elementos que um processo recebe ao chegar no tribunal, que o identificam), enquanto todos os demais, que já existem no acervo, viriam com suas covariáveis preenchidas.

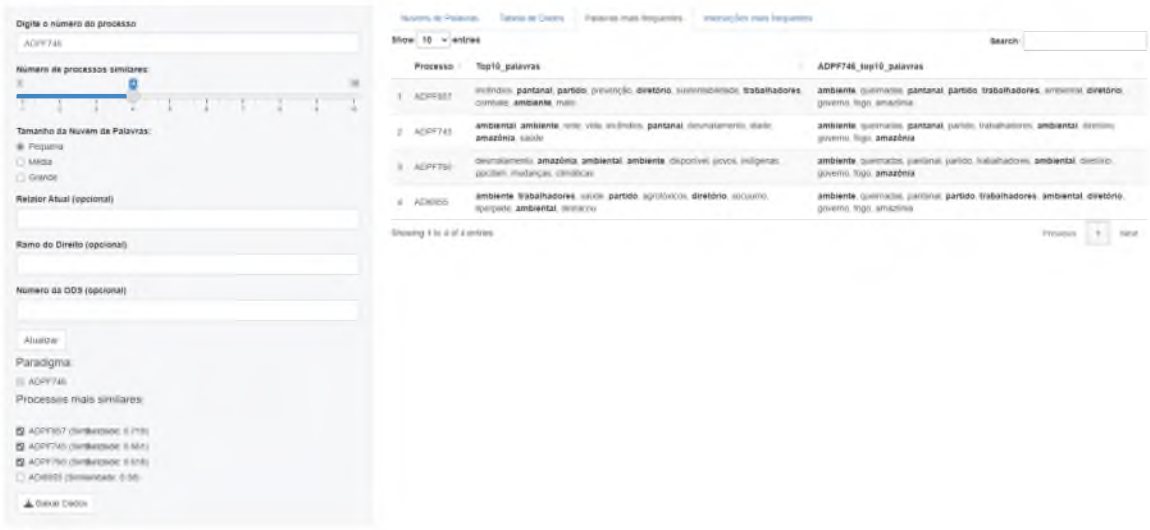

Figura 16: Terceira aba da aplicação — palavras mais frequentes.

Na terceira aba da aplicação, apresentada na Figura 16, é informado para cada processo as 10 palavras mais frequentes após aplicação de um pesado filtro de *stopwords*, a fim de evitar palavras pouco úteis para entendimento do contexto do processo pareadas com as 10 palavras mais frequentes do processo paradigma. Pretende-se assim fornecer ao usuário mais um elemento de identificação visual de padrões e semelhanças entre os processos. As palavras que são comuns entre o processo que se está analisando e o processo paradigma são destacadas em negrito, facilitando ainda mais a interpretação do usuário.

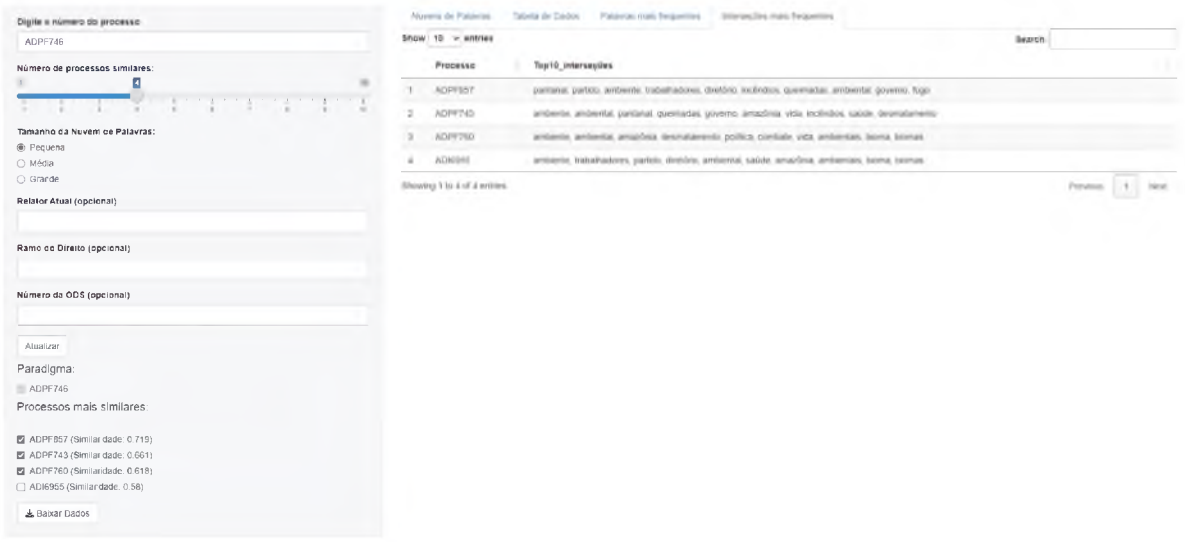

Figura 17: Quarta aba da aplicação Interseções mais frequentes.

Na quarta e última aba da aplicação, vista na Figura 17, são apresentadas ao usuário as 10 palavras com interseção mais frequente entre o processo paradigma e o processo que se está analisando, fornecendo mais uma ferramenta de identificação da semelhança do conteúdo do processo. Nesta etapa também é realizada uma rotina de remoção de diversas palavras vazias, a fim de evitar palavras com pouco poder comparativo para este contexto.

Por fim, o usuário tem a opção de baixar os dados contidos na aplicação, permitindo a continuidade do trabalho por fora do aplicativo. Como a ferramenta mais comum de trabalho no tribunal para este tipo de insumo é o *Excel*, os dados são exportados em formato *xlsx*, sempre nomeados por padrão *dados_processo_DATAHOJE.xlsx*, mas é facultado ao usuário a possibilidade de alterar o nome.

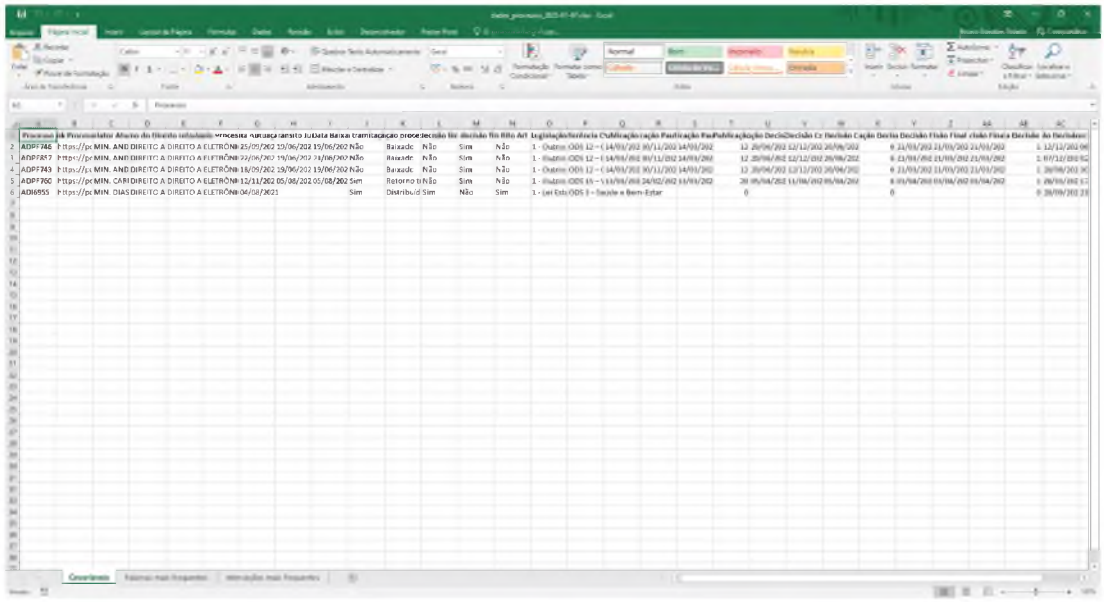

Figura 18: Forma dos dados exportados da aplicação.

Ao abrir os dados baixados em um aplicativo externo para visualização de arquivos em formato *xlsx*, ilustrado na Figura 18, o usuário terá o conteúdo das abas da aplicação referentes a lista dos processos, palavras mais frequentes e interseções mais frequentes, sem o elemento visual da nuvem de palavras e sem o corpo textual da petição inicial, possibilitando assim a continuidade do trabalho após análise visual realizada dentro da aplicação.

A Figura 19 trás um exemplo da eficácia da aplicação. Na parte de cima, nota-se que, fixado o processo ADPF 746 como paradigma e solicitando os dois processos mais similares, tem-se a seleção dos processos ADPF 857 e ADPF 743. Na parte de baixo da colagem, observa-se a comparação com o documento que fora utilizado para validação, disponível em Informação à sociedade — ADPFs 743, 746 e 857. Neste caso, observa-se que a aplicação foi bem sucedida em acertar os dois processos que de fato foram considerados como os mais similares pelo Tribunal.

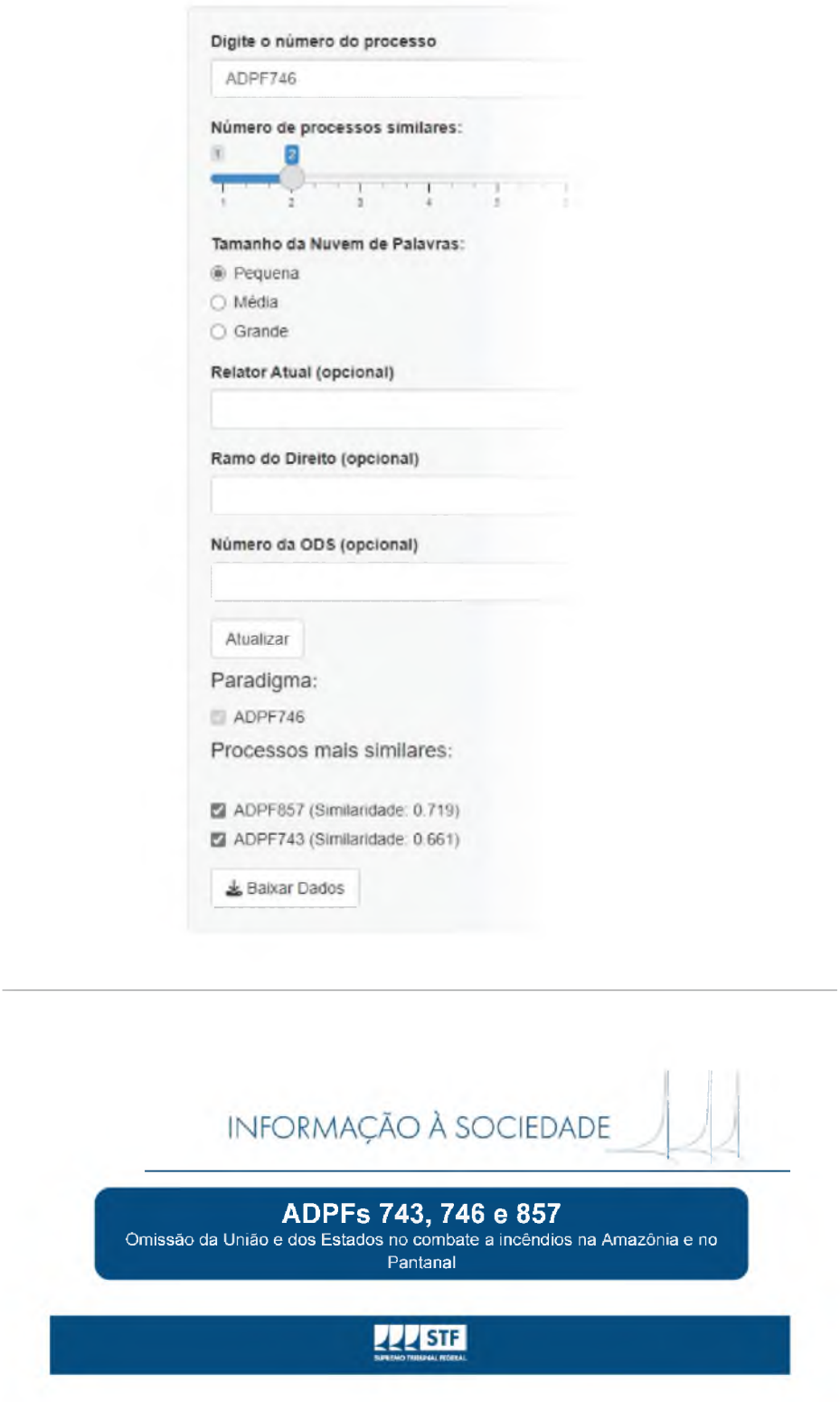

Figura 19: Colagem de *print* da aplicação com *print* de informação à sociedade.

Como a única fonte de dados para a modelagem é a petição inicial do processo, esta aplicação torna possível um pré encaminhamento de processos recém chegados ao tribunal. Antes mesmo de serem extraídas e registradas dimensões pertinentes para análise processual, como o ramo do direito a que ele se refere, assunto do processo e demais informações, de forma que para a constante atualização desta aplicação é necessário somente a inserção do novo documento ocerizado na base de dados, e a remoção de processos já baixados da mesma.

5 Conclusão

Esse trabalho teve como objetivo contribuir para a meta de gestão da Presidência do Ministro Luís Roberto Barroso de reduzir em 20% o acervo de processos de controle concentrado do Supremo Tribunal Federal. Para tanto, foram empregadas técnicas de Processamento de Linguagem Natural com o intuito de identificar similaridades entre os processos do acervo. A identificação automatizada de processos similares visa reduzir o tempo e o esforço manual demandados para essa análise. Adicionalmente, essa abordagem pode facilitar o trâmite desses processos perante o Tribunal, contribuindo assim para a meta da gestão.

Para efetuar esta busca de similaridades, utilizou-se as técnicas de vetorização *Bag of Words* e *doc2vec*, combinadas com medidas de similaridade estatística como distância do cosseno e divergência de Jensen-Shannon. Construiu-se um aplicativo *Shiny* para efetuar esta busca na rotina de trabalho do Tribunal de forma contínua e automatizada.

O modelo *doc2vec* para vetorização de textos do Tribunal se mostrou inferior na qualidade de vetorização ante a uma vetorização mais simples produzida pelo método *Bag of Words* no contexto deste trabalho. Dentre as medidas de distância estatística testadas, a que melhor performou foi a distância do cosseno. Dos casos estudados, a vetorização e a métrica escolhidas para a aplicação final detectaram os verdadeiros processos mais similares, figurando entre o 1º a 4º mais similar. Acredita-se que estas técnicas mediram com fidedignidade as semelhanças do conteúdo textual das petições iniciais dos processos, tornando assim possível o reconhecimento de determinados padrões no acervo.

A utilização do *t-SNE* para visualização das semelhanças dos processos não mostrou-se efetiva, desfigurando as semelhanças esperadas. Por outro lado, a utilização de nuvens de palavras mostrou-se efetiva para fornecer uma noção preliminar do conteúdo das petições iniciais, possibilitando comparação visual entre o conteúdo de dois processos.

O aplicativo *Shiny* produzido nesse trabalho fornecerá aos analistas do Tribunal uma ferramenta simples e útil — porém com bom desempenho — para a busca de processos similares no acervo de controle concentrado.

Referências

ARTES, R.; BARROSO, L. P. *Métodos multivariados de análise estatística.* São Paulo: Blucher, 2023.

BOLUKBASI, T. et al. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. 2016. Disponível em: ⟨https://doi.org/10.48550/arXiv.1607.06520⟩.

BURBEA, J.; RAO, C. On the convexity of some divergence measures based on entropy functions. *IEEE Transactions on Information Theory*, v. 28, n. 3, p. 489–495, 1982.

DROST, H. Philentropy: Information theory and distance quantification with r. *Journal of Open Source Software*, v. 3, n. 26, p. 765, 2018. Disponível em: ⟨https://joss.theoj.org/papers/10.21105/joss.00765⟩.

EVERITT, B.; SKRONDAL, A. *The cambridge dictionary of statistics.* Cambridge: Cambridge University Press, 2010. v. 4.

FIRTH, J. R. *Studies in linguistic analysis.* 8ª. ed. [S.l.]: Blackwell, Oxford, 1962.

FREITAS, L. J. G. Clusterização de textos aplicada ao tratamento de dados jurídicos desbalanceados. *Tese (Mestrado em Estatística) – Departamento de estatística, Universidade de Brasília*, p. 16, 2023.

FREITAS, L. J. G.; ALENCAR, E.; RODRIGUES, T. C. V. Rafa 2030-deep learning applied to brazilian supreme court legal documents and un 2030 agenda. Galoá, 2022.

FREITAS, L. J. G. et al. Catboost algorithm application in legal texts and un 2030 agenda. *Revista de Informatica Teórica e Aplicada - RITA - ISSN 2175-2745. Vol. 30, Num. 02 (2023) 51-58*, 2023.

FREITAS, L. J. G. et al. Text clustering applied to data augmentation in legal contexts. *arXiv preprint arXiv:2404.08683*, 2024.

GHORBANI, H. Mahalanobis distance and its application for detecting multivariate outliers. *Facta Universitatis, Series: Mathematics and Informatics*, p. 583–595, 2019.

HINTON, G. E.; ROWEIS, S. Stochastic neighbor embedding. MIT Press, v. 15, 2002. Disponível em: ⟨https://proceedings.neurips.cc/paper_files/paper/2002/file/6150ccc6069bea6b5716254057a194ef-Paper.pdf⟩.

JOHNSON, R. A.; WICHERN, D. W. *Applied Multivariate Statistical Analysis.* 6. ed. Essex: Pearson, 2007.

JOTHEESWARAN, J.; KUMARASWAMY, Y. Opinion mining using decision tree based feature selection through manhattan hierarchical cluster measure. *Journal of Theoretical & Applied Information Technology*, v. 58, n. 1, 2013.

JURAFSKY, D.; MARTIN, J. H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models.* 3. ed. Prentice-Hall, 2025. Disponível em: ⟨https://web.stanford.edu/~jurafsky/slp3⟩.

LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, Ieee, v. 86, n. 11, p. 2278–2324, 1998.

LU, J.; HENCHION, M.; NAMEE, B. M. Diverging divergences: Examining variants of jensen shannon divergence for corpus comparison tasks. 2021.

MAATEN, L. Van der; HINTON, G. Visualizing data using t-sne. *Journal of machine learning research*, v. 9, n. 11, 2008.

MOHAMMADHASSANZADEH, H. et al. Using natural language processing to examine the uptake, content, and readability of media coverage of a pan-canadian drug safety research project: Cross-sectional observational study. *JMIR formative research*, JMIR Publications Inc., Toronto, Canada, v. 4, n. 1, p. e13296, 2020.

MORETTIN, P. A.; SINGER, J. M. *Estatística e Ciência de Dados*. 1ª. ed. Rio de Janeiro: LTC, 2023.

NAHRSTEDT, F. et al. An empirical study on the energy usage and performance of pandas and polars data analysis python libraries. In: *Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering*. New York, NY, USA: Association for Computing Machinery, 2024. (EASE '24), p. 58–68. ISBN 9798400717017. Disponível em: ⟨https://doi.org/10.1145/3661167.3661203⟩.

NUNES, M. Análise de sentimentos com o r: Bojack horseman vs brooklyn nine-nine. 2019. Disponível em: ⟨https://marcusnunes.me/posts/analise-de-sentimentos-com-r-bojack-horseman-vs-brooklyn-99/⟩. Acesso em: 31 de dez. de 2024.

QADER, W. A.; AMEEN, M. M.; AHMED, B. I. An overview of bag of words; importance, implementation, applications, and challenges. In: IEEE. *2019 international engineering conference (IEC)*. [S.l.], 2019. p. 200–204.

RICARDO, B.-Y.; BERTHIER, R.-N. *Modern information retrieval: the concepts and technology behind search*. [S.l.]: New Jersey, USA: Addi-son-Wesley Professional, 2011.

ROSS, S. D. Ensaios: Remover acentos e desenvolver lematização no r. 2020. Disponível em: ⟨https://rpubs.com/StevenDuttRoss/lemmatizacao-no-R⟩. Acesso em: 31 de dez. de 2024.

SARICA, S.; LUO, J. Stopwords in technical language processing. *Plos one*, Public Library of Science San Francisco, CA USA, v. 16, n. 8, p. e0254937, 2021.

SILVA, T. *Racismo algorítmico: inteligência artificial e discriminação nas redes digitais*. 1. ed. São Paulo: Edições Sesc SP, 2022. ISBN 978-65-86111-70-5.

SOARES, A. Introdução à análise textual aplicada à sociologia. 2022. Disponível em: ⟨https://soaresalisson.github.io/analisetextual/⟩. Acesso em: 31 de dez. de 2024.

TVERSKY, A. Features of similarity. *Psychological review*, American Psychological Association, v. 84, n. 4, p. 327, 1977.

VEROUTIS, P.; FAJARDO, E. Markov chain monte carlo methods in cryptography. 2021.