Instituto Federal de Educação, Ciência e Tecnologia da Paraíba

Campus João Pessoa

Programa de Pós-Graduação em Tecnologia da Informação

SAMUEL DE AGUIAR RODRIGUES

INTEGRAÇÃO DE ALGORITMO DE ASSOCIAÇÃO COM ALGORITMO DE CLASSIFICAÇÃO: EXPERIÊNCIA NO JUIZADO ESPECIAL CÍVEL DA PARAÍBA

DISSERTAÇÃO DE MESTRADO

JOÃO PESSOA

2022

Samuel de Aguiar Rodrigues

Integração de Algoritmo de Associação com Algoritmo de Classificação: Experiência no Juizado Especial Cível da Paraíba

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Tecnologia da Informação, pelo Programa de Pós- Graduação em Tecnologia da Informação do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – IFPB.

Orientador: Profa. Dra. Juliana Dantas Ribeiro Viana de Medeiros

Coorientador: Prof. Dr. Francisco Dantas Nobre Neto

João Pessoa

2022

Dados Internacionais de Catalogação na Publicação – CIP
Biblioteca Nilo Peçanha – IFPB, *campus* João Pessoa

R696i Rodrigues, Samuel de Aguiar.
Integração de algoritmo de associação com algoritmo de classificação : experiência no Juizado Especial Cível da Paraíba / Samuel de Aguiar Rodrigues. – 2022.
73 f. : il.

Dissertação (Mestrado em Tecnologia da Informação) – Instituto Federal da Paraíba – IFPB / Programa de Pós-Graduação em Tecnologia da Informação - PPGTI.
Orientadora: Profª. Dra. Juliana Dantas R. Viana de Medeiros.
Coorientador: Prof. Dr. Francisco Dantas Nobre Neto.

1. Algoritmo de associação. 2. Algoritmo de classificação. 3. Mineração de texto jurídico. 4. Inteligência artificial. 5. Aprendizado de máquina. 6. Juizado Especial Cível da Paraíba. I. Título.

CDU 004.421

Bibliotecária responsável Taize Araújo da Silva – CRB15/536

Samuel de Aguiar Rodrigues

Integração de Algoritmo de Associação com Algoritmo de Classificação: Experiência no Juizado Especial Cível da Paraíba

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Tecnologia da Informação, pelo Programa de Pós-Graduação em Tecnologia da Informação do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – IFPB.

Aprovado em 31 de outubro de 2022.

BANCA EXAMINADORA:

gov.br Documento assinado digitalmente
FRANCISCO PETRONIO ALENCAR DE MEDEI
Data: 03/01/2023 14:40:58-0300
Verifique em https://verificador.iti.br

Prof. Dr. Francisco Petronio Alencar de Medeiros – IFPB Avaliador

gov.br Documento assinado digitalmente
ALISSON VASCONCELOS DE BRITO
Data: 06/01/2023 08:29:01-0300
Verifique em https://verificador.iti.br

Prof. Dr. Alisson Vasconcelos de Brito – UFPB Avaliador Externo

gov.br Documento assinado digitalmente
Juliana Dantas Ribeiro Viana de Medeiros
Data: 28/12/2022 16:18:15-0300
Verifique em https://verificador.iti.br

gov.br Documento assinado digitalmente
Francisco Dantas Nobre Neto
Data: 29/12/2022 15:53:22-0300
Verifique em https://verificador.iti.br

Profa. Dra. Juliana Dantas Ribeiro Viana de Medeiros (Orientador)

Prof. Dr. Francisco Dantas Nobre Neto (Coorientador)

Visto e permitida a impressão
João Pessoa

gov.br Documento assinado digitalmente
DAMIRES YLUSKA DE SOUZA FERNANDES
Data: 10/01/2023 09:30:26-0300
Verifique em https://verificador.iti.br

Profa. Dra. Damires Yluska de Souza Fernandes – Coordenador PPPGTI

Este trabalho é dedicado ao meu pai,

Hélio Rodrigues da Silva,

que faleceu dias antes do meu ingresso no mestrado
e não pôde vivenciar esta minha evolução

acadêmica e profissional.

AGRADECIMENTOS

Agradeço a Deus pela oportunidade, sabedoria e força para prosseguir com esta pesquisa, iluminando minha mente em cada avanço. À minha mãe, Marizene Delgado, e a minha avó, Clarina Maria de Aguiar, por toda dedicação, suporte, incentivo e zelo. Por cada oração a Deus em meu favor para conclusão desta pesquisa.

Ao meu pai (*in memoriam*), Hélio Rodrigues da Silva, por cada exemplo de dedicação e perserverança nos objetivos. Pela educação e incentivos constantes em cada passo dado no crescimento pessoal e profissional.

À minha família pela paciência e compreensão nas ausências e dedicação necessária, em especial à minha irmã, Helizene Nívea, pelo apoio e incentivo na concretização dos meus objetivos.

À minha namorada e doutoranda, Jessica Santos, pelo apoio, compreensão e incentivo constante nas atualizações e pesquisas desenvolvidas.

À distinta orientadora por sua infinita paciência, compreensão, suporte e disponibilidade no acompanhamento desta pesquisa. Ao coorientador pela paciência, atenção e colaboração nas revisões e melhorias do trabalho.

Aos meus amigos do mestrado, M.Sc Adelson Barreto e Jonathas Eiras, sempre presentes no andamento deste curso.

Ao meu amigo e atual Diretor de Tecnologia da informação, Ney Robson Pereira de Medeiros, pela compreensão e colaboração para concretização da pesquisa. Ao meu amigo e colega de trabalho, M.Sc Fabrício Araújo, pelo apoio e compartilhamento do vasto conhecimento acadêmico que possui. À minha amiga, Dra. Julietty Santos, pelo apoio e incentivo, mesmo à distância, na conclusão deste trabalho.

Ao Tribunal de Justiça da Paraíba, na pessoa do atual Diretor de Tecnologia da Informação, assim como, do diretor anterior José Teixeira de Carvalho Neto, por todo o suporte necessário para o desenvolvimento, evolução e acompanhamento da pesquisa.

A todos que contribuíram de forma direta ou indireta na concretização desta pesquisa.

RESUMO

O acesso ao Poder Judiciário ainda é restrito a quem detém conhecimentos mínimos sobre o papel de advogados, defensoria pública e o ingresso através do juizado especial. Este último, permite que um cidadão acesse o judiciário independente da contratação de advogado, através do instituto do *jus postulandi*. Tal direito, possibilita ao cidadão a criação de um documento direcionado ao juiz para protocolo do processo que contém o ocorrido e os pedidos, denominado petição inicial. No Tribunal de Justiça da Paraíba, o sistema para protocolar processo judicial eletrônico é o PJe. Nele, além da petição inicial no momento do protocolo, também são exigidas outras informações, a exemplo da classe e assunto judicial, dois conceitos jurídicos pouco conhecidos para a maior parte da população. A presente pesquisa busca eliminar a barreira do conhecimento jurídico sobre classe e assunto judicial, provendo mecanismos de inteligência artificial através de aprendizado de máquina para possibilitar que um processo seja classificado automaticamente no assunto judicial correspondente ao conteúdo narrado na petição inicial para a classe judicial Procedimento do Juizado Especial Cível. Para atingir este objetivo, foi desenvolvido um modelo de Inteligência Artificial a partir de dados extraídos e processados considerando as regras da legislação aplicável. Posteriormente os dados foram preparados e submetidos ao algoritmo de associação *Apriori* gerando insumos para o algoritmo de classificação *SVM*. O modelo foi avaliado alcançando 77,94% de acurácia e 77,35% de *F1 Score*, sendo submetido à validação cruzada na configuração *10-fold Cross-Validation*, permanecendo estável com variação positiva em mais de 2%. Este modelo visa facilitar a criação de novas portas de entrada ao judiciário que permitam ao cidadão preencher minimamente o ocorrido e suas informações pessoais para acionar o judiciário na criação de um processo de pequenas causas.

Palavras-chaves: mineração de texto, *apriori*, algoritmo de associação, *svm*, algoritmo de classificação, mineração de texto jurídico, inteligência artificial, aprendizado de *máquina, Pje, jus postulandi*.

ABSTRACT

Access to the Judiciary is still restricted to those who have minimal knowledge about the role of lawyers, public defenders, and admission through the special court. The latter allows a citizen to access the judiciary independent of hiring a lawyer, through the *jus postulandi* institute. This right allows the citizen to create a document directed to the judge for the protocol of the process that contains what happened and the requests, called initial petition. In the Court of Justice of Paraíba, the system for filing electronic judicial proceedings is the PJe. In addition to the initial petition at the time of the protocol, other information is also required, such as the judicial class and judicial matter, two legal concepts that are little known to most of the population. The present research seeks to eliminate the barrier of legal knowledge about judicial class and judicial matter, providing artificial intelligence mechanisms through machine learning to enable a process to be automatically classified in the judicial subject corresponding to the content narrated in the initial petition for the judicial class Special civil court. To achieve this objective, an Artificial Intelligence model was developed from data extracted and processed considering the rules of the applicable legislation. Subsequently, the data were prepared and submitted to the Apriori association algorithm, generating inputs for the SVM classification algorithm. Finally, the model was evaluated reaching 77.94% of accuracy and 77.35% of F1 Score, cross-validated in the 10-fold Cross-Validation configuration, remaining stable with positive variation in more than 2%. This model aims to facilitate the creation of new gateways to the judiciary that allows citizens to fill in what happened minimally and their personal information to trigger the judiciary in creating a small claims process.

Keywords: text mining, apriori, association algorithm, svm, classification algorithm, legal text mining, artificial intelligence, machine learning, PJe, jus postulandi.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AM	Aprendizagem de máquina
Art.	Artigo
ASCII	*American Standard Code for Information Interchange*
BERT	*Bidirectional Encoder Representations from Transformers*
Bp	Bloco de petição
CNJ	Conselho Nacional de Justiça
FA	Filtro de Assunto
FP	Filtro de Petição
GA	Grupo de Assunto
HTML	*HyperText Markup Language*
IA	Inteligência Artificial
Nº	Número
NLTK	*Natural Language Toolkit*
PDF	*Portable Document Format*
PJe	Processo Judicial eletrônico (Sistema)
STJ	Superior Tribunal de Justiça
SVM	*Support Vector Machine*
TF-IDF	*Term Frequency–Inverse Document Frequency*
TJPB	Tribunal de Justiça da Paraíba
TJRO	Tribunal de Justiça de Rondônia
TJTO	Tribunal de Justiça do Tocantins
TPU	Tabela Processual Unificada
TRF4	Tribunal Regional Federal da 4ª Região

SUMÁRIO

SUMÁRIO

1. INTRODUÇÃO

Com a democratização do acesso à internet, muitos serviços para o cidadão podem ser acessados de casa, sem que seja necessário o deslocamento às instalações físicas das instituições públicas e privada. Contudo, quando estamos tratando do acesso ao Judiciário, nem sempre é possível solucionar uma situação sem a necessidade de deslocamento ao fórum para que o problema possa ser analisado e tenha sua tratativa iniciada.

Com a pandemia da COVID19, as esferas governamentais iniciaram esforços na tentativa de tornar o máximo de serviços públicos acessíveis de forma remota através de aplicativos ou aplicações WEB, diminuindo assim a quantidade de pessoas aglomeradas em prédios públicos para tratar problemas de baixa complexidade (CÂMARA DOS DEPUTADOS, 2022).

No judiciário não foi diferente, uma vez que diversas iniciativas foram impulsionadas, a exemplo do Juízo 100% Digital (CNJ, 2022), Balcão Virtual (CNJ, 2022) e formulários eletrônicos de atermação (TJPB, 2022). Este último visa solucionar o problema do deslocamento do cidadão ao fórum para narrar os fatos ocorridos contra determinada pessoa que o tenha ofendido, utilizando mecanismos de conciliação e evitando que um processo judicial seja protocolado.

1.1. Problema

De acordo com o art. 9º da Lei nº 9.099/95, todo cidadão tem o direito de peticionar no judiciário – ato de solicitar algo judicialmente – através do *jus postulandi*. Tal direito permite que o cidadão formalize um processo judicial sem a necessidade de contratação de um advogado. Contudo, a formalização de um processo judicial ainda requer conhecimentos jurídicos que são desconhecidos pela maior parte da população.

Apesar da digitalização de vários serviços e dos esforços que o judiciário tem feito para evoluir a prestação dos serviços oferecidos à população, o acesso à Justiça ainda tem um longo caminho a ser percorrido para que se torne efetivo no dia a dia do cidadão. Para Gagno e Bufon (2020): “É impossível se cogitar um Estado de Direito sem a realização do direito fundamental de acesso à justiça, sendo assim, para que todos se beneficiem das leis é indispensável que todos possam servir-se delas.”

Atualmente, o maior sistema de processo judicial eletrônico em uso no judiciário brasileiro é o PJe[1]. Instituído pela Resolução nº 185/2013, este sistema foi desenvolvido e ampliado de forma colaborativa entre os servidores do judiciário, permitindo que todo processo judicial seja acompanhado e evoluído eletronicamente, com o uso de certificação e assinatura digital para a validade dos documentos produzidos. Neste sistema, para protocolar um processo, o usuário deve preencher diversos campos que caracterizam e classificam a sua demanda para que assim, quando o processo for cadastrado, o fluxo de trabalho a ser aplicado seja adequado à situação narrada na petição inicial. Dos campos obrigatórios a serem preenchidos no protocolo de um processo, três campos são considerados essenciais para a classificação do processo: petição inicial, classe judicial e assunto judicial.

A petição inicial é o documento que descreve todos os fatos ocorridos entre as partes – pessoas envolvidas no processo – junto ao pedido direcionado ao juiz que analisará e decidirá sobre a questão. A classe judicial, por sua vez, é o campo que define o rito processual que será aplicado na evolução daquele processo e o assunto judicial define o tema abordado no processo. Com base nesses campos, o judiciário define a melhor forma de tratar o processo internamente, buscando solucionar o problema da forma mais célere possível em conformidade com a legislação vigente. (CNJ, 2014)

Para o cidadão comum protocolar um processo sem auxílio, por mais simples que seja, é necessário conhecer minimamente estes campos, para que sua demanda seja tratada adequadamente. Também é necessário o uso de um certificado digital para assinatura dos documentos e acompanhamento do processo.

Nesse contexto, o presente trabalho busca propor um aperfeiçoamento ao PJe, mediante uma funcionalidade concebida com o uso de um modelo de aprendizado de máquina.

Dessa forma, o cidadão comum poderá protocolar um processo sem a necessidade de conhecer classe ou assunto judicial. Isso é possível pois a classe é delimitada pelo tipo de protocolo e o assunto é predito pelo modelo. Assim, é solucionado o problema da necessidade de conhecimento jurídico para acessar a justiça.

1.2. Objetivos

1.1.1. Objetivo geral

Construir um modelo utilizando técnicas de aprendizado de máquina para identificação automática do assunto judicial, a partir da petição inicial, visando facilitar o acesso do cidadão à Justiça no âmbito dos Juizados Especiais Cíveis.

---

[1] Tribunal de Justiça da Paraíba, 2022. Sistema PJe, 1º Grau. Disponível em <https://pje.tjpb.jus.br/pje>. Acesso em 20 de agosto de 2022

Com base neste contexto, a seguinte questão de pesquisa se apresenta:

- Como integrar algoritmo de associação com algoritmo de classificação para identificar assunto judicial com base na petição inicial?

1.1.2. Objetivos específicos

Os tópicos a seguir são os objetivos específicos desta pesquisa:

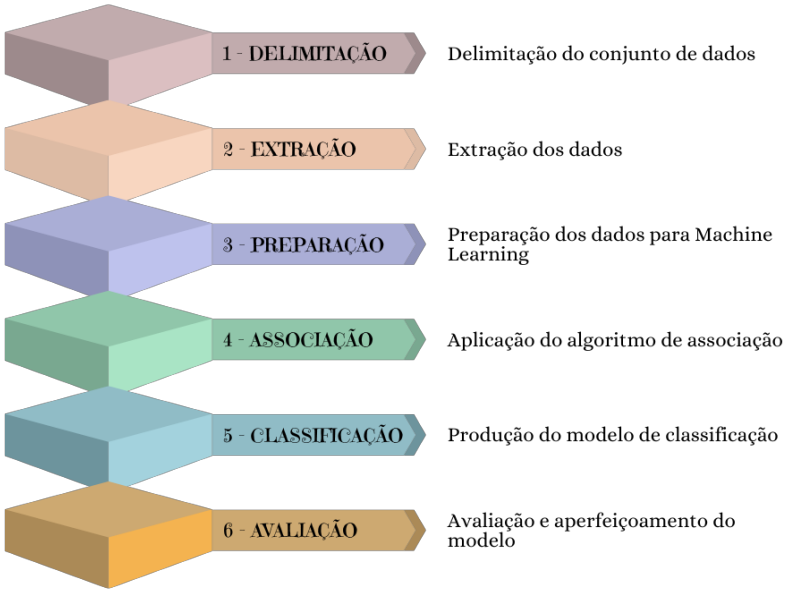
- Definir critérios de extração e extrair petições iniciais passíveis de uso pelo modelo;
- Preparar texto das petições para o aprendizado de máquina;
- Definir critérios de extração e extrair assuntos judiciais com exemplos para treinamento do modelo;
- Identificar associações de palavras de acordo com o assunto judicial;
- Produzir um modelo de classificação em conjunto com as associações identificadas;
- Verificar o desempenho do modelo gerado em comparação ao modelo sem uso de associações;
- Avaliar a consistência da performance do modelo.

1.3. Metodologia da Pesquisa

Para atingir os objetivos anteriormente descritos, definimos uma questão de pesquisa exploratória (EASTERBROOK, SINGER, *et al.*, 2008) que se propõe a investigar sobre como integrar algoritmo de associação com algoritmo de classificação para identificar assunto judicial com base na petição inicial.

A pesquisa foi conduzida em seis etapas, conforme resumido a seguir na Figura 1.

Figura 1 – Metodologia Aplicada com as respectivas etapas da pesquisa.



Fonte: Desenvolvido pelo autor.

- **Delimitação do conjunto de dados:** Nesta etapa, foi delimitado o escopo do modelo gerando diretrizes para guiar a execução das próximas etapas quanto aos limites dos dados que podem ser utilizados.
- **Extração de dados:** Nesta etapa, extraímos os dados em conformidade com a legislação vigente e as diretrizes recebidas como entrada. Foram aplicados diversos filtros que reduziram o conjunto de dados com a remoção de dados irrelevantes. Com isso geramos como saída para a próxima etapa o *corpus* com os dados brutos das petições iniciais para ser preparado e utilizado na execuçao dos algoritmos.
- **Preparação dos dados:** Na etapa de prepação dos dados, recebemos como entrada o *corpus* em sua forma primária e eliminamos dados irrelevantes para o treinamento do modelo. Os dados foram padronizados e tokenizados juntamente com a aplicação de técnicas como a extensão e remoção das *stopwords*. Como resultado geramos um *corpus* com as petições iniciais transformadas em *tokens* selecionados e seus respectivos assuntos.
- **Aplicação do algoritmo de associação:** Nesta etapa, preparamos os *tokens* do *corpus* recebido na preparação e aplicamos o algoritmo de associação por conjunto de documentos de cada assunto judicial. Posteriormente aplicamos uma máscara composta dos *tokens* identificados pelo algoritmo de associação para selecionar os *tokens* a serem utilizados na criação do modelo de classificação.

- **Produção do modelo de classificação:** Nesta etapa, geramos dois modelos, sendo um com base na saída da etapa de preparação dos dados e outro com base na saída da etapa de aplicação do algoritmo de associação. Os modelos gerados foram utilizados como entrada para etapa de avaliação.
- **Avaliação do modelo:** Utilizamos a técnica validação cruzada para validar a performance dos dois modelos gerados na etapa anterior, produzindo um comparativo das métricas avaliadas com seus respectivos ganhos de performance.

Para avaliação do módulo, optamos por uma análise quantitativa. Os modelos gerados foram avaliados isoladamente para verificacao de sua performance de acurácia e *F1 Score* (JAIN e JAIN, 2021). Posteriormente aplicamos uma validacão cruzada visando verificar a confiabilidade na avaliação isolada.

1.4. Aplicabilidade

O resultado principal desta pesquisa é o desenvolvimento de um modelo de IA que, analisando a petição inicial possa identificar o assunto judicial correspondente, para permitir que um cidadão comum possa preencher minimamente o relato do ocorrido e suas informações pessoais para acionar o judiciário na criação de um processo de pequenas causas no PJe, que é o sistema utilizado pelo Tribunal de Justiça da Paraíba.

Entretanto, tem-se a expectativa que o modelo gerado possa ser aprimorado para ser utilizado no judiciário nacional. Devido ao fato de o PJe ser um sistema colaborativo, os tribunais podem submeter melhorias que são avaliadas por revisores espalhados em todo país.

Importante ressaltar que a pesquisa tem sido construída considerando todos os aspectos necessários para implantação do modelo, incluindo conformidade com a legislação atual na extração e processamento de dados.

Além da possibilidade de adoção nacional, o CNJ tem incentivado a implementação de modelos de inteligência artificial, possibilitando que o modelo seja utilizado nacionalmente ou utilizado como base de implementação para integração de outros modelos.

1.5. Estrutura do Documento

O documento está organizado da seguinte forma:

- O capítulo 1 trata da introdução, contextualizando a temática em que a pesquisa está inserida e os objetivos que pretende alcançar;
- O capítulo 2 apresenta a fundamentação teórica dos conteúdos abordados na pesquisa, além de uma breve análise dos trabalhos que possuem alguma relação com esta pesquisa;

- O capítulo 3 trata da concepção do modelo através de um tratamento de dados detalhado e sua medição de performance;
- No capítulo 4 tratamos dos resultados do modelo criado, efetuando uma validação cruzada para validar a performance alcançada;
- Por fim, no capítulo 5 concluímos o trabalho com considerações sobre os resultados obtidos e indicando possíveis caminhos de continuação da pesquisa.

2. FUNDAMENTAÇÃO TEÓRICA

Desde 2003, o Conselho Nacional de Justiça (CNJ) publica um relatório anual de análise e diagnóstico do Poder Judiciário denominado *Justiça em Números*, além de disponibilizar um painel virtual com os dados utilizados. O painel aponta que nos últimos 3 anos foram protocolados mais de 13 milhões de processos com a classe Procedimento do Juizado Especial Cível, além de evidenciar um crescimento de mais de 37% na quantidade de processos em 2021 contra pouco mais de 3% de crescimento em 2020 (CNJ, 2022).

2.1. Processo

Na justiça brasileira os conflitos de diversas naturezas são resolvidos através de processos, nos quais o Estado aplica a lei para garantir o direito dos envolvidos.

"*Processo: Instrumento mediante o qual o Estado soluciona conflitos através da aplicação da lei; série ordenada de atos necessários e assinalados em lei para que se investigue, para que se esclareça a controvérsia e, afinal, para que se solucione a pendência.*" (STJ, 2016).

Popularmente, o processo é difundido como uma ação judicial. Neste sentido o glossário do STJ conceitua ação como: "*Meio processual pelo qual o cidadão pode buscar uma decisão judicial para, através de advogado constituído nos autos, fazer valer um direito que acredita ser-lhe assegurado pela ordem jurídica.*" (STJ, 2016).

Desta forma, para fins deste documento, adotaremos o conceito de processo como o instrumento para solução de conflitos que possui, entre outros atributos, petição inicial, classe e assunto.

2.2. Petição inicial

De acordo com Theodoro Junior (2016), o Estado só pode agir quando provocado pela parte interessada sobre sua demanda. Para isso, o veículo de manifestação formal da demanda é a petição inicial. A petição inicial consiste no primeiro requerimento dirigido a uma autoridade judiciária para que seja iniciado ou provocado um litígio (STJ, 2016).

Podemos considerar, então, que a petição inicial é o documento que descreve os fatos ocorridos entre os envolvidos, direcionado à autoridade judiciária – juiz – com o respectivo pedido feito pela parte interessada/demandante.

2.3. Classificação Processual

O CNJ instituiu a Resolução nº 46/2007, posteriormente alterada pela Resolução nº 326/2020 que cria as Tabelas Processuais Unificadas (TPUs) do Poder Judiciário.

> "*Art. 1º Ficam criadas as Tabelas Processuais Unificadas do Poder Judiciário, objetivando a padronização e uniformização taxonômica e terminológica de classes, assuntos, movimentação e documentos processuais no âmbito da Justiça Estadual, Federal, do Trabalho, Eleitoral, Militar da União, Militar dos Estados, do Superior Tribunal de Justiça e do Tribunal Superior do Trabalho, a serem empregadas em sistemas processuais, cujo conteúdo, disponível no Portal do Conselho Nacional de Justiça (www.cnj.jus.br), integra a presente Resolução. (Redação dada pela Resolução nº 326, de 26.6.2020)*" (CNJ, 2022).

Com a implantação desta resolução, que visa padronizar os dados de classificação dos processos e assim gerar dados estatísticos precisos, todos os Tribunais do País passaram a adotar a classificação unificada em seus processos.

De acordo com o CNJ (2014), a classe pode ser definida como o procedimento judicial ou administrativo adequado ao pedido. E o assunto engloba as matérias ou temas discutidos no processo. De acordo com o glossário disponível no Sistema de Gestão de Tabelas Processuais Unificadas (SGT), a classe Procedimento do Juizado Especial Cível, código 436, por exemplo, é aplicável a todas as ações ajuizadas nos Juizados Especiais e que observem o rito especial das Leis 9.099/95 (Justiça Estadual) e 10.259/01 (Justiça Federal). Já a classe Crimes Ambientais, código 293, é aplicável em ações fundadas nos tipos previstos na Lei 9.605/1998. Desta forma, podemos ter processos configurados como:

- Processo A
    - Classe – Procedimento do Juizado Especial Cível (436)
    - Assunto – Perdas e Danos (7698)
- Processo B
    - Classe – Crimes Ambientais (293)
    - Assunto – Dano Ambiental (10438)

Podemos observar que enquanto a classe está mais relacionada a legislação aplicável e seus ritos, o assunto define o tema específico do processo.

A presente pesquisa está focada na classificação dos processos em assuntos judiciais. Neste sentido, vejamos o que diz o Manual de Utilização das TPUs:

"*O pedido com as suas especificações bem como os fatos e fundamentos jurídicos serão analisados pelo cadastrador para definir o assunto principal da lide, que deverá ser o primeiro assunto cadastrado*." (CNJ, 2014).

2.4. Aprendizado de Máquina

O Aprendizado de Máquina (AM) é um subconjunto da Inteligência Artificial que se utiliza de dados disponíveis em seu ambiente para aprender com a experiência, e usa-os para melhorar o seu desempenho (VIEIRA, 2022 apud LATAH, 2018). O uso de aprendizado de máquina tem sido incentivado no judiciário através do programa Justiça 4.0 que, dentre outras iniciativas, busca tornar o judiciário brasileiro mais acessível, disponibilizando novas tecnologias e inteligência artificial.

Desta forma, o momento torna-se oportuno para implantar modelos de IA, a exemplo do que está sendo produzido com a presente pesquisa, pois há fundamentação e incentivo do Conselho Nacional de Justiça junto aos Tribunais brasileiros.

Geron (2019) define aprendizado de máquina, ou *machine learning*, como a ciência e a arte da programação de fazer computadores aprenderem com os dados.

O aprendizado de máquina originou-se da necessidade de um cientista estadunidense jogar damas contra um computador. Arthur Lee Samuel, ao perceber que sempre ganhava do programa que havia feito, programou para que o computador aprendesse com os jogos anteriores. A partir de então, perdeu várias vezes da máquina. (ALENCAR, 2022). Para Samuel (1969), "Programar computadores para aprender com a experiência deve eventualmente eliminar a necessidade de grande parte desse esforço de programação detalhada.".

2.4.1. Algoritmos Supervisionados

Segundo Faceli, Lorena, *et al.* (2011), o termo supervisionado vem da simulação da presença de um supervisor externo, que conhece o rótulo desejado para cada exemplo utilizado no treinamento e passa a ter a capacidade de avaliar o rótulo de novos exemplos. Tal descrição se encaixa apropriadamente no que buscamos nesta pesquisa, pois, de acordo com Faceli, Lorena, *et al.* (2011), o aprendizado supervisionado pode ser dito como preditivo, englobando os algoritmos de classificação.

2.4.1.1. Algoritmos de classificação

De acordo com Izbicki e Dos Santos (2020), algoritmos de classificação tratam problemas cuja variável de resposta é qualitativa, a exemplo de um classificador automático de dígitos escritos à mão com base em imagens previamente analisadas. Faceli, Lorena, *et al.* (2011) complementam definindo como uma função que, dado um conjunto de exemplos rotulados com valores de um domínio conhecido, constrói um estimador; caso o domínio seja infinito ou desconhecido, teremos um problema de regressão. Tarefas de classificação são comumente associadas aos algoritmos

mais populares, a exemplo do *Support Vector Machine* (*SVM*), *Random Forest*, *Naive Bayes*, *K-nearest-neighbors* (*KNN*) e *Logistic Regression*.

Quando a classificação está relacionada a textos, o *SVM* tem se destacado dos demais. Chatterjee *et al*. (2019) utilizaram *SVM* para classificar textos de várias fontes em conjunto com uma abordagem multithreading utilizando GPU/CUDA para aumentar a velocidade do processo, após comparar com os algoritmos *Decision Tree, Random Forest e Naive Bayes* confirmando o melhor desempenho do *SVM*. Tegegnie *et al*. (2017) utilizaram *SVM* para classificar várias notícias de categorias diferentes, utilizando uma abordagem hierárquica para redução das *features* genéricas resultando em um aumento da acurácia do modelo.

Clavié e Alphonsus (2021) utilizaram *SVM* na classificação de textos legais demonstrando uma performance competitiva com modelos de *deep learning* pré-treinandos no campo do processamento de linguagem natural. Ni *et al.* (2016) compararam *KNN* e *SVM* na classificação de textos no campo petroquímico objetivando encontrar os melhores parâmetros para classificar documentos e identificou melhor performance do algoritmo *SVM* em todos os cenários analisados.

Wang *et al.* (2021) analisaram respostas em texto para identificar personalidade comparando *SVM*, **KNN**, *XGboost, Naive Bayes* e *Logistic Regression* na classificação dos textos. Nos resultados *SVM* e *Naive Bayes* se destacaram na performance, tendo o *SVM* demonstrado mais estabilidade que o *Naive Bayes*.

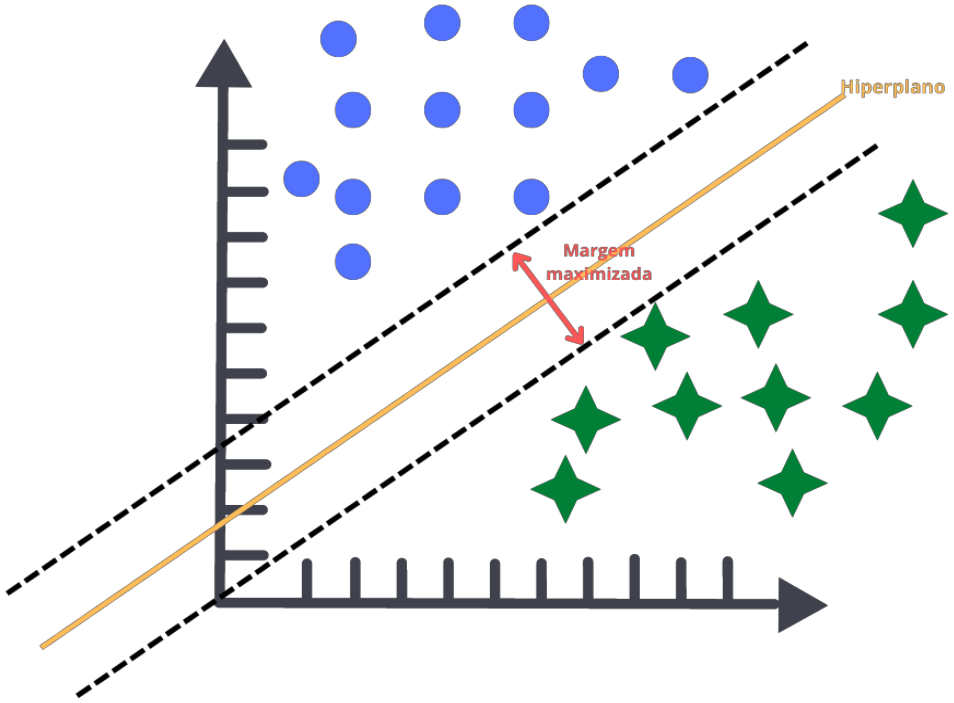
Por se tratar de um problema de classificação de texto e considerando o destaque do *SVM* para este tipo de tarefa, nesta pesquisa foi escolhido o algoritmo *Support Vector Machine* para classificação.

2.4.1.1.1. *Support Vector Machine*

Support Vector Machine (*SVM*) classifica os dados encontrando o hiperplano que maximiza a margem entre as classes nos dados de treinamento (ALBON, 2018).

Isto significa que, dado um conjunto de dados com duas classes, o algoritmo procura encontrar os pontos que estão nas bordas dos grupos de cada classe e cria um hiperplano no meio, de forma a maximizar as margens. A Figura 2 ilustra a situação.

Figura 2 – Ilustração do funcionamento do *SVM*.



Fonte: Desenvolvido pelo autor.

As margens representam a distância do hiperplano para a borda e é utilizada como base para definir se um novo dado se aproxima mais de um grupo que de outro. "Quando existem muitos hiperplanos que separam os dados perfeitamente, o *SVM* busca aquele que tem a maior margem M, isto é, aquele que fica mais distante de todos os pontos observados. Os pontos utilizados para definir as margens são chamados de vetores de suporte." (IZBICKI e DOS SANTOS, 2020).

2.4.2. Algoritmos não supervisionados

Aprendizagem de Máquina não supervisionada envolve reconhecimento de padrões sem que haja um rótulo previamente conhecido. Todas as variáveis usadas na análise são usadas como entrada, motivo pelo qual é uma técnica recomendada em mineração de dados para clusterização e associação (ALLOGHANI *et al*., 2020).

Segundo Geron (2019), as técnicas de *k-Means* e *Apriori* estão entre os mais importantes algoritmos não supervisionados.

2.4.2.1. Algoritmos de associação

Regras de associação é um ramo da mineração que tem sido estudado com sucesso em diversas áreas e tem como objetivo principal extrair associações com base na frequência dos itens (MAHMOOD, SHAHBAZ e GUERGACHI, 2014) .

Geron (2019) cita como exemplo a descoberta, no registro de vendas de uma empresa, de associação entre produtos, ou seja, quem compra produto A tende a comprar produto B. Krisnanto et al. (2022) utilizaram o algoritmo *Apriori* para melhorar a estratégia de marketing e vendas analisando as transações para identificar produtos que seriam promovidos e recomendados aos consumidores.

2.4.2.1.1. *Apriori*

Faceli, Lorena, *et al.* (2011) relatam que *Apriori* foi o primeiro algoritmo de regras de associação. Buscando conjuntos de itens frequentes varrendo o banco de dados, uma transação por vez, incrementando o suporte de todos os itens envolvidos naquela transação.

A relevância da regra de associação pode ser identificada através de dois parâmetros: suporte mínimo (porcentagem da combinação de itens em todas as transações) e confiança mínima (força do relacionamento entre os itens de uma regra de associação) (KRISNANTO *et al.*, 2022).

A associação se dá por uma transação A → B, em que A U B é considerado um conjunto frequente de itens. A confiança da associação é a relação entre o número de transações que incluem todos os itens de A para o número de transações que incluem todos os itens de B, enquanto o suporte relativo do conjunto de itens é a divisão do número total de itens do conjunto pelo número total de transações (FACELI, LORENA, *et al.*, 2011).

Apriori tem sido muito utilizado em sistemas de recomendação pela sua característica de relacionar um item com outro. Guo, Wang, Li (2017) utilizaram o *Apriori* para melhorar o sistema de recomendação de um *e-commerce mobile*, aumentando a eficiência e a acurácia das recomendações feitas em tempo real.

No campo da saúde, Ma *et al*. (2022) utilizaram o algoritmo para identificar padrões de doenças que poderiam estar relacionadas em um mesmo indivíduo. Após execução e filtros das associações encontradas, identificou 110 combinações de doenças que possuem uma forte relação, evidenciando a importância e a utilidade do uso do *Apriori* para estudos de múltiplas comorbidades.

Por outro lado, o algoritmo *Apriori* também tem sido usado para melhoria na seleção de *features* que serão utilizadas nos algoritmos de aprendizado de máquina. Neste sentido, Jain e Jain (2021) utilizaram a descoberta de regras de associação com o algoritmo *Apriori* para melhorar a acurácia do *SVM*, *Naive Bayes, Random Forest* e *Logistic Regression*, obtendo melhoria de 4 a 6% nas acurácias investigadas para uso na classificação de sentimentos. Li e Yao (2019) utilizaram

o algoritmo *Apriori* em conjunto com o *k-medoids* para criar uma camada adicional na seleção de *features*, combinando os itens frequentes identificados para melhorar a performance de sua classificação. Althuwaynee *et al*. (2021) chegaram a utilizar *Apriori* para identificar *features* não rotuladas com base em fatores derivados da topografia observada nos mapas estudados após utilizar t-SNE para redução das *features*.

Nesta pesquisa utilizamos *Apriori* para otimizar a seleção de *features*.

2.5. Pré-processamento dos dados

Uma etapa comum no contexto de processamento de linguagem natural, antes de que os dados sejam submetidos aos algoritmos de aprendizado de máquina, é aquela referente ao pré-processamento dos dados, visando uniformizar as palavras que serão processadas pelo algoritmo. De acordo com Sarica e Luo (2021), para assegurar acurácia e eficiência das tarefas de processamento de linguagem natural, como classificação de textos, palavras com pouca relevância semântica, frequentemente chamadas de *stopwords*, precisam ser removidas antes do processamento.

Para Cho, Lee e Kang (2021), a representação dos dados em um espaço vetorial é fundamental para tarefas de processamento de linguagem natural, notadamente, classificação de texto. Eles relatam que muitos pesquisadores têm conduzido diferentes métodos de representar as unidades de palavras, chamadas de *tokens*. De acordo com eles, a técnica de tokenização é o simples método de decompor uma sentença por espaços em branco.

Ao unir as duas técnicas, podemos remover os *tokens* que pertencem ao conjunto de palavras das *stopwords*. Ainda de acordo com Makrehchi e Kamel (2017), *stopwords* são as chamadas palavras comuns, ruídos, palavras irrelevantes e palavras não discriminativas, que podem ter uma frequência de ocorrência elevada, mas pouca relevância semântica. De acordo com eles, tais palavras podem ser divididas em dois grupos: gerais e específicas de um domínio. As palavras gerais estão disponíveis em domínio amplo, enquanto que as palavras específicas de um domínio formam um conjunto de palavras com valor discriminativo para o domínio em que é aplicado.

2.6. Métricas de Avaliação

Os modelos de classificação supervisionada são avaliados por sua capacidade de sugerir o rótulo correto para o item analisado de acordo com o rótulo previamente estabelecido. O resultado de uma sugestão pode ser Falso Positivo (FP), Verdadeiro Positivo (*True Positive* - TP), Verdadeiro Negativo (*True Negative* - TN) e Falso Negativo (FP). A partir desses resultados, é possível criar uma matriz de avaliação denominada Matriz de Confusão. Com base nesses resultados é possível avaliar a performance do modelo através das métricas de Acurácia

(*Accuracy*), Precisão (*Precision*), Revocação (*Recall*) e *F1 Score* (*F-measure*). (JAIN e JAIN, 2021)

Figura 3 – Fórmulas das métricas de avaliação.

Evaluation Metric	Formula
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
Precision (P)	$\frac{TP}{TP + FP}$
Recall(R)	$\frac{TP}{TP + FN}$
F-measure	$2\frac{P.R}{P + R}$

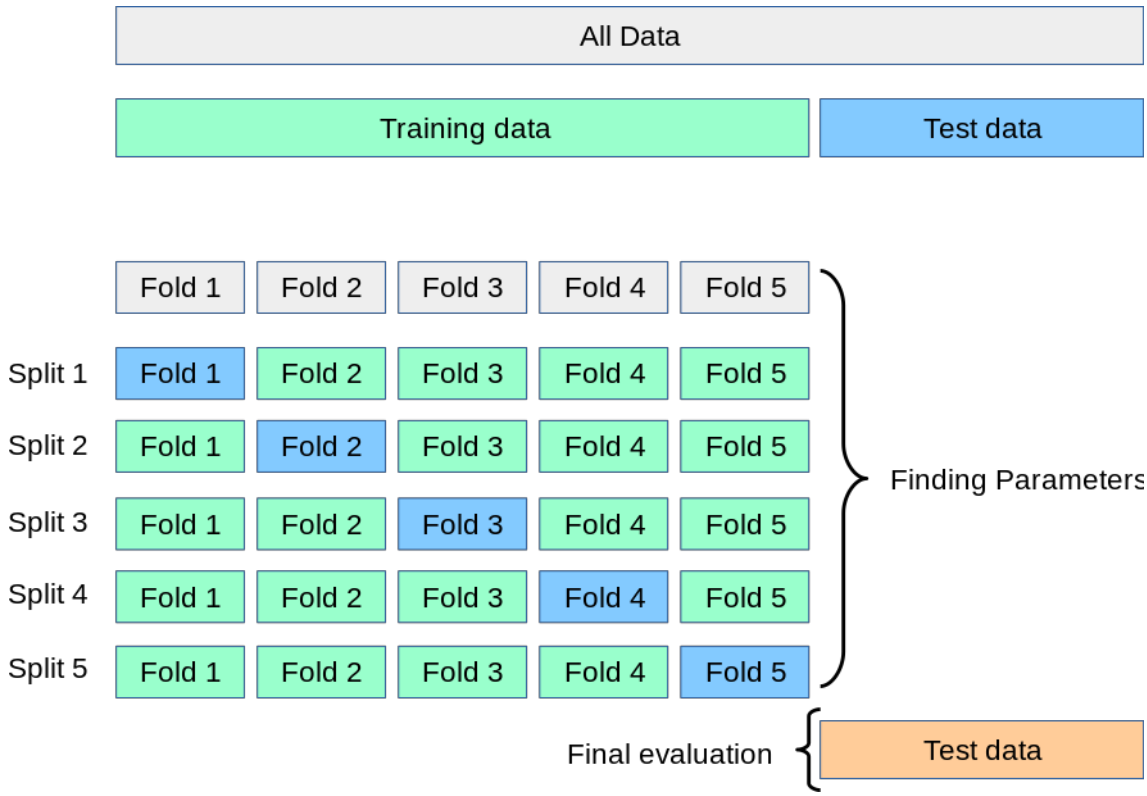
Fonte: Guo, Wang, Li (2017)

A Figura 3 demonstra as fórmulas das métricas com base nos resultados das sugestões. Nesta pesquisa adotaremos a acurácia e o *F1 Score* como métricas principais, tendo em vista o *F1 Score* relacionar a precisão e a revocação.

2.6.1. K-fold Cross-Validation

A técnica *K-fold Cross-Validation* busca aumentar a qualidade do modelo através de iterações de treinamento e teste do modelo, evitando a sobreotimização do modelo supervisionado de AM. De acordo com SciKit-Learn (2022), o conjunto de treino é dividido em K subconjuntos, o modelo é treinado utilizando K-1 subconjuntos e validado utilizando o subconjunto restante. O processo é repetido até que todos os subconjuntos sejam utilizados como conjunto de validação. A performance é medida pela média das avaliações obtidas nas iterações do *loop*. A Figura 4 ilusta o processo em que K é igual a 5.

Figura 4 – *5-fold Cross-Validation.*



Fonte: SciKit-Learn (2022)

Para a criação dos subconjuntos é necessário dividir o *corpus* em K partes iguais. Na biblioteca do *scikitlearn* a classe *KFold* é responsável por esse processo, contudo não há uma preocupação da quantidade de amostras que são direcionadas para uma classe em detrimento de outra. Para solucionar esse problema, a classe *StratifiedKFold* pode ser utilizada, pois divide o número de amostras igualmente entre as classes na criação do subconjunto, evitando *corpus* desbalanceados na execução dos treinamentos.

Considerando a característica desbalanceada do conjunto de dados desta pesquisa, optamos por utilizar a classe *StratifiedKFold* para execução da técnica.

2.7. Trabalhos Relacionados

Quando se trata de IA no Judiciário, é importante considerar a legislação que afeta a situação analisada. As iniciativas de IA foram crescendo rapidamente no judiciário brasileiro e o CNJ, sempre atento às mudanças e evoluções que ocorrem nos tribunais, editou a resolução nº 332/2020 que dispõe sobre a ética, a transparência e a governança na produção e no uso de IA no Poder Judiciário. Neste sentido, a Fundação Getúlio Vargas tem promovido junto ao Judiciário, com a coordenação do Ministro Luis Felipe Salomão do Superior Tribunal de Justiça (STJ), um relatório sobre as inteligências artificiais aplicadas neste poder, além dos projetos que ainda estão em desenvolvimento nos tribunais.

De acordo com a FGV (2020), dos 63 projetos de IA analisados no Judiciário em 2020, 29 estavam em desenvolvimento, 7 em fase piloto e 27 em produção. Desses projetos, podemos destacar a iniciativa do Tribunal Regional Federal da 4ª região (TRF4) na análise de processos para sugerir ao servidor retificar o assunto judicial caso o resultado encontrado fosse divergente do cadastrado. Diferentemente desta pesquisa, cujo foco é o PJe, o sistema de processo eletrônico envolvido nessa iniciativa é o eProc, além da análise e sugestão ser efetuada após o processo já estar em andamento. Na segunda fase do relatório esse projeto não foi listado.

Podemos destacar também o Projeto TUA, em desenvolvimento pelo STJ, que visa identificar os assuntos dos processos eletrônicos do Tribunal. Assim como a iniciativa do TRF4, não se aplica ao PJe e a análise também é feita após o protocolo do processo. Outros modelos focados em identificar os assuntos judiciais são citados no relatório no campo "Outros sistemas e inteligência artificial em desenvolvimento", mas sem maiores detalhes.

O relatório foi ponto de partida para uma pesquisa mais aprofundada sobre o assunto. No ano de 2022 foi publicada a segunda fase da pesquisa feita pelo Centro de Inovação, Administração e Pesquisa do Judiciário da FGV. De acordo com a FGV (2022), o número de iniciativas cresceu para 64, contudo vale ressaltar que da primeira fase para a segunda algumas iniciativas foram substituídas.

Além das iniciativas anteriormente citadas, podemos destacar o projeto Toth cujo objetivo é recomendar classes e assuntos com base na petição inicial. Essa iniciativa se aproxima um pouco mais da presente pesquisa, pois trata também do PJe. Contudo, sua aplicação é genérica e aplicada após o protocolo do processo, enquanto esta pesquisa se propõe a se especializar apenas em processos de Juizado Especial Cível através dos assuntos relacionados à classe Procedimento do Juizado Especial Cível. A iniciativa informa no relatório que a base foi suficiente para algumas classes e assuntos mais relevantes e que houve insuficiência de petições para atender à totalidade de classes e assuntos do CNJ, característica da genericidade proposta.

Outra iniciativa que abrange classificação de assuntos é o Peticionamento Inteligente do Tribunal de Justiça de Rondônia (TJRO). Essa iniciativa permite que delegacias enviem documentos escaneados para serem convertidos em processos. O sistema identifica as informações necessárias, entre elas o assunto criminal, para protocolar no PJe. A iniciativa se restringe ao núcleo criminal e relata que a maioria dos modelos de assuntos não foram possíveis pela baixa quantidade de exemplos.

Temos ainda a iniciativa do Tribunal de Justiça de Tocantins (TJTO) denominada Sistema de Classificação de Petições Judiciais. Tal iniciativa visa melhorar a assertividade das informações dos processos através da sugestão de classe e assunto com base na TPU. Igualmente ao TRF4, o projeto está vinculado ao EPROC e em fase de implantação.

Outras iniciativas são citadas no relatório, mas sem relação com a presente pesquisa. Contudo, apresenta um panorama geral do uso de IA no Judiciário com informações importantes sobre o uso de técnicas e tecnologias aplicadas no desenvolvimento das iniciativas.

Outros trabalhos também têm sido desenvolvidos com contexto jurídico, ainda que fora do Judiciário. Silveira *et al*. (2021) utilizaram um modelo pré-treinado com casos dos Estados Unidos para avaliar seu uso em conjunto com outras técnicas de IA e posteriormente validar com especialistas se os temas identificados estariam corretos. Para atingir esse objetivo, processaram os parágrafos individualmente adicionando legislação nas citações encontradas e criando clusters para identificar as palavras mais relevantes. Como resultado, 84,6% dos tópicos identificados foram validados pelos especialistas. Aguiar *et al*. (2021) aplicaram técnica similar, mas no contexto da Justiça Brasileira; utilizando um conjunto de petições de 6 diferentes classes judiciais fornecidas pelo Tribunal de Justiça do Ceará (TJCE) em conjunto com o modelo BERT (BERTimbau - modelo pré-treinado em português do Brasil) atingindo 0.88 de *F1 Score* macro. Para seleção das petições analisadas, se restringiram no uso apenas das petições que possuíam legislação citada. Diferente dessa última pesquisa, utilizamos apenas uma classe judicial e focamos na correta identificação do assunto a ser relacionado no processo; também não há restrição quanto a citação de legislação de forma obrigatória, mesmo porque a classe judicial objeto da presente pesquisa tem como público o cidadão sem conhecimento jurídico.

Aguiar *et al*. (2022) refinaram a pesquisa anterior utilizando os melhores cenários que haviam identificado. Com isso, conseguiu atingir 0.89 de *F1 Score* macro, mas como manteve o mesmo contexto e conjunto de dados, os pontos divergentes da presente pesquisa permaneceram.

Sousa (2019) realizou um trabalho em conjunto com uma equipe formada por magistrados e analistas judiciários do Tribunal de Justiça do Tocantins (TJTO) para classificar os assuntos judiciais dos processos de juizado especial cível da comarca de Augustinópolis. Sua pesquisa foi feita utilizando PDFs de petições iniciais extraídas do EPROC e após análise comparativa de algoritmos identificou que o *SVM* seria o mais adequado para o cenário. Seu projeto foi intitulado MinerJus e atingiu acurácia de 93,58% em sua predição. Enquanto Sousa (2019) atuou nas petições em PDF do EPROC, a presente pesquisa é direcionada ao PJe e utilizou apenas documentos HTML para criação do modelo. Vale destacar, ainda, que, ao contrário do MinerJus, as petições extraídas do PJe abrangem todo o Estado da Paraíba e são submetidas a uma solução de algoritmo não supervisionado aliada a um algoritmo supervisionado.

3. CONCEPÇÃO DO MODELO

Um conjunto de técnicas de mineração de texto e aprendizado de máquina foram combinadas para gerar um modelo de IA que pudesse sugerir o assunto judicial principal de um processo judicial analisando o conteúdo de sua petição inicial. Este capítulo apresenta o detalhamento do processo de seleção dos dados até a criação do modelo de classificação.

3.1. Delimitação do conjunto de dados

A pesquisa foi iniciada a partir da definição das diretrizes macro a serem usadas no modelo, que foram o guia das tomadas de decisão quanto aos refinamentos e processamentos dos dados.

O PJe do Tribunal de Justiça da Paraíba possui, hoje, mais de 300 classes ativas no primeiro grau de jurisdição, contudo o escopo da presente pesquisa foi delimitado na classe Procedimento do Juizado Especial Cível, código 436.

A escolha da classe judicial Procedimento do Juizado Especial Cível foi motivada pelo fato de ser a única classe atualmente disponível para protocolo de *jus postulandi* nos Juizados Especiais Cíveis do Tribunal de Justiça da Paraíba, que é o foco e a fonte de dados desta pesquisa. Esta classe é responsável hoje por mais de 8% dos processos em andamento no TJPB, o que corresponde a mais de 71 mil processos e é a responsável por abranger as demandas popularmente conhecidas como pequenas causas.

As demandas de pequenas causas possuem valor da causa limitado e focam em problemas mais simples, de baixa complexidade, geralmente abrangidas pelo código de defesa do consumidor ou pequenos desentendimentos, em conformidade com a Lei nº 9099/95 que em seu Art. 9º define que: “Nas causas de valor até vinte salários mínimos, as partes comparecerão pessoalmente, podendo ser assistidas por advogado; nas de valor superior, a assistência é obrigatória.”.

A classe Procedimento do Juizado Especial Cível está disponível apenas no primeiro grau de jurisdição, motivo pelo qual nossas fontes de dados são os assuntos judiciais e as petições iniciais do PJe, primeiro grau, do Tribunal de Justiça da Paraíba associados a esta classe.

No PJe, cada processo inicia com uma petição no formato PDF ou HTML, sendo frequente o uso dos dois formatos; comumente o segundo é usado apenas para informar que o conteúdo da petição está em formato PDF. Optamos por trabalhar apenas com as petições em HTML cujo conteúdo completo consta neste formato, por produzirem insumos suficientes para a pesquisa e por sua disponibilidade imediata em banco de dados.

Considerando a característica da proposta de uso em ambiente real do modelo gerado, optamos por utilizar apenas dados associados aos assuntos principais passíveis de utilização em Julho de 2022.

Por fim, evitando a geração de um modelo excessivamente genérico e de baixa performance, devido a alta granularidade da relação de classes e assuntos das Tabelas Processuais Unificadas, elencamos as diretrizes usadas como guia da pesquisa:

1. Dados do sistema PJe do Tribunal de Justiça da Paraíba;
2. Apenas dados do primeiro grau de jurisdição;
3. Apenas dados da classe Procedimento do Juizado Especial Cível;
4. Apenas dados de petições iniciais em HTML;
5. Apenas dados associados a processos cujo assunto principal está apto para utilização em Julho de 2022.

3.2. Extração dos dados

Com as diretrizes macro definidas sobre a utilização dos dados, iniciamos a extração buscando a forma mais eficiente de extrair os dados considerando cada detalhe necessário para produção de um conjunto de dados que representasse casos reais válidos para serem usados no treinamento e posterior predição. A extração foi executada diretamente na base de dados PostgreSQL do sistema PJe.

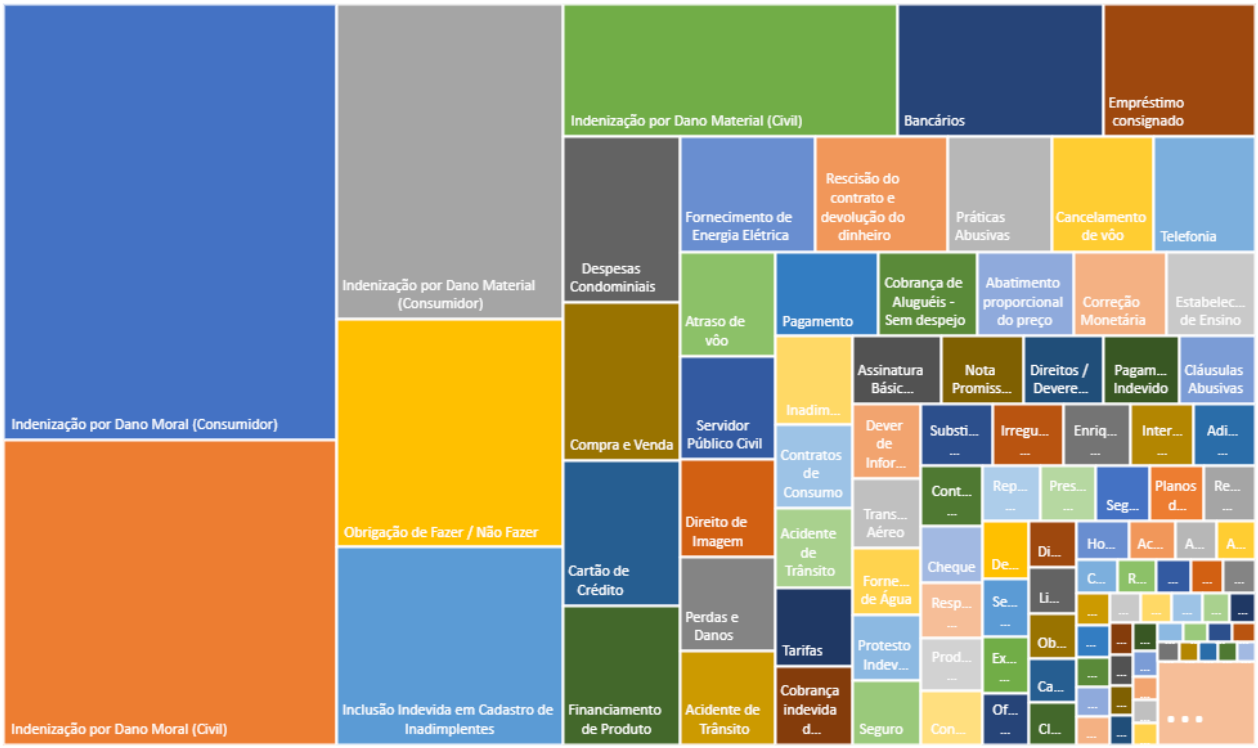
O fato da legislação brasileira estar em constante atualização, torna a atividade de seleção dos dados jurídicos, a serem utilizados na criação do modelo, uma atividade fundamental para manter a coerência com a legislação vigente no momento da extração.

3.2.1. Assuntos judiciais

Os assuntos judiciais são o grande foco desta pesquisa, pois eles são os rótulos utilizados pelo modelo para classificar os textos das petições. Para que o resultado da pesquisa pudesse ser implementável, foi importante considerar algumas regras na seleção dos assuntos que compõem o treinamento.

Para nosso estudo, separamos todos os assuntos aplicados em processos cuja petição foi incluída em formato HTML, gerando um conjunto de dados de 205 assuntos aplicados em 457.343 processos, em conformidade com a diretriz 4 previamente definida. A Figura 5 ilustra o quantitativo de processos por assunto em relação ao todo.

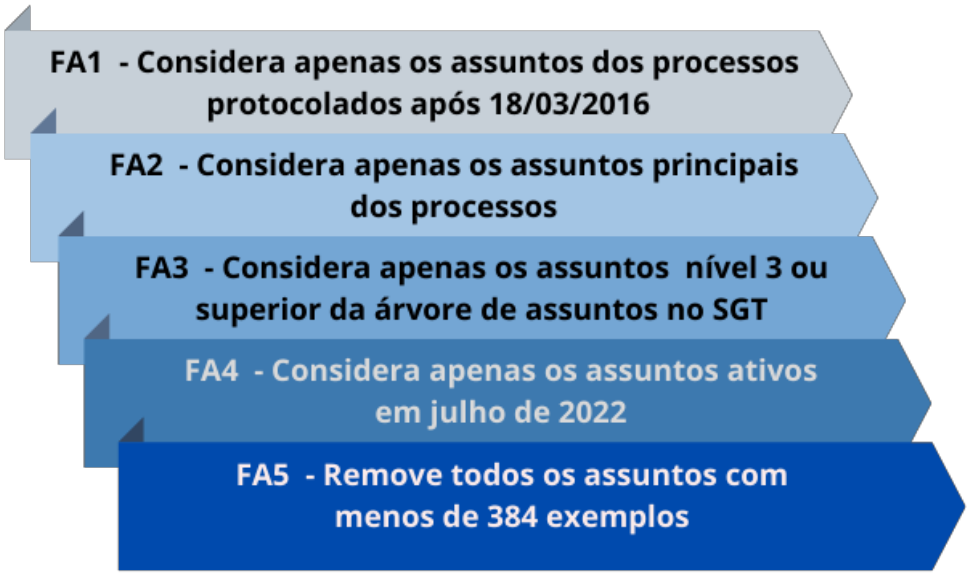
Figura 5 – Mapa de Árvore antes dos filtros (Processos por assunto).



Fonte: Desenvolvido pelo autor.

Em seguida, foram aplicados alguns filtros na consulta SQL utilizada na extração para compatibilizar o conjunto de dados com as regras utilizadas atualmente no sistema e na legislação. A Figura 6 ilustra os filtros utilizados.

Figura 6 – Lista de filtros aplicados nos assuntos.



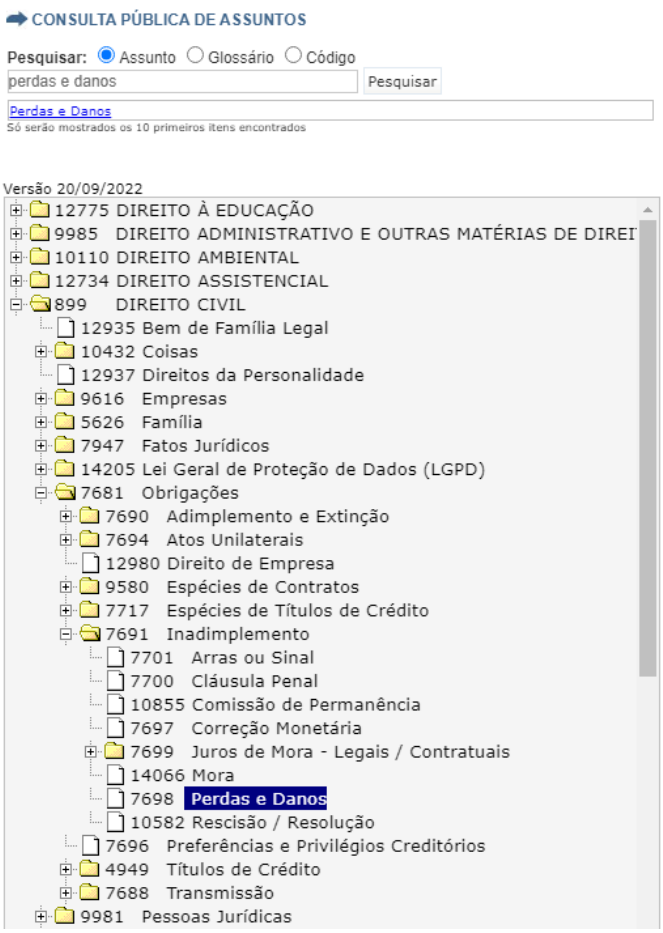
Fonte: Desenvolvido pelo autor.

O primeiro filtro, FA1, considera apenas processos protocolados após 18 de março de 2016, data que entrou em vigor o Código de Processo Civil de 2015 (STJ, 2022), reduzindo para 192 assuntos aplicados em 341.478 processos.

O sistema trabalha com dois tipos de assuntos: os principais e os complementares. Os assuntos principais representam da melhor forma o tema do processo em questão e são obrigatórios, enquanto que os assuntos complementares são auxiliares e de indicação opcional. A partir disso, foi aplicado o filtro FA2 passando a considerar apenas os assuntos principais dos processos; com isso reduzimos ainda mais a quantidade de processos, restando 177 assuntos aplicados em 203.706 processos, ou seja, menos da metade da quantidade inicial de processos.

Os assuntos judiciais da Tabela Processual Unificada estão organizados no SGT de forma hierárquica implementando o conceito de árvore, em que um assunto raiz possui seus filhos até atingir o nível de folha. A Figura 7 ilustra o assunto folha "Perdas e Danos" selecionado no sistema.

Figura 7 – Consulta Pública de assuntos



Fonte: Sistema de Gestão de Tabelas Processuais Unificadas[2].

[2] Conselho Nacional de Justiça, 2022. Sistema de Gestão de Tabelas Processuais Unificadas. Disponível em: <https://www.cnj.jus.br/sgt/consulta_publica_assuntos.php>. Acesso em: 20 de setembro de 2022.

Com base nas diretrizes implementadas pelo Datajud, editado pela Portaria nº 160/2020 (CNJ, 2022), visando à padronização dos dados do judiciário para uso estatístico e de tomada de decisão, apenas assuntos nível 3 ou superior na árvore de assuntos judiciais da Tabela Processual Unificada devem ser utilizados nos processos. Em conformidade com essa diretriz, aplicamos o filtro FA3 removendo todos os processos em que os assuntos principais não atendiam ao requisito.

Em seguida, foi aplicado o filtro FA4, que remove todos os assuntos inativos em julho de 2022, tornando o treinamento mais eficiente apenas com os assuntos possíveis no período da execução.

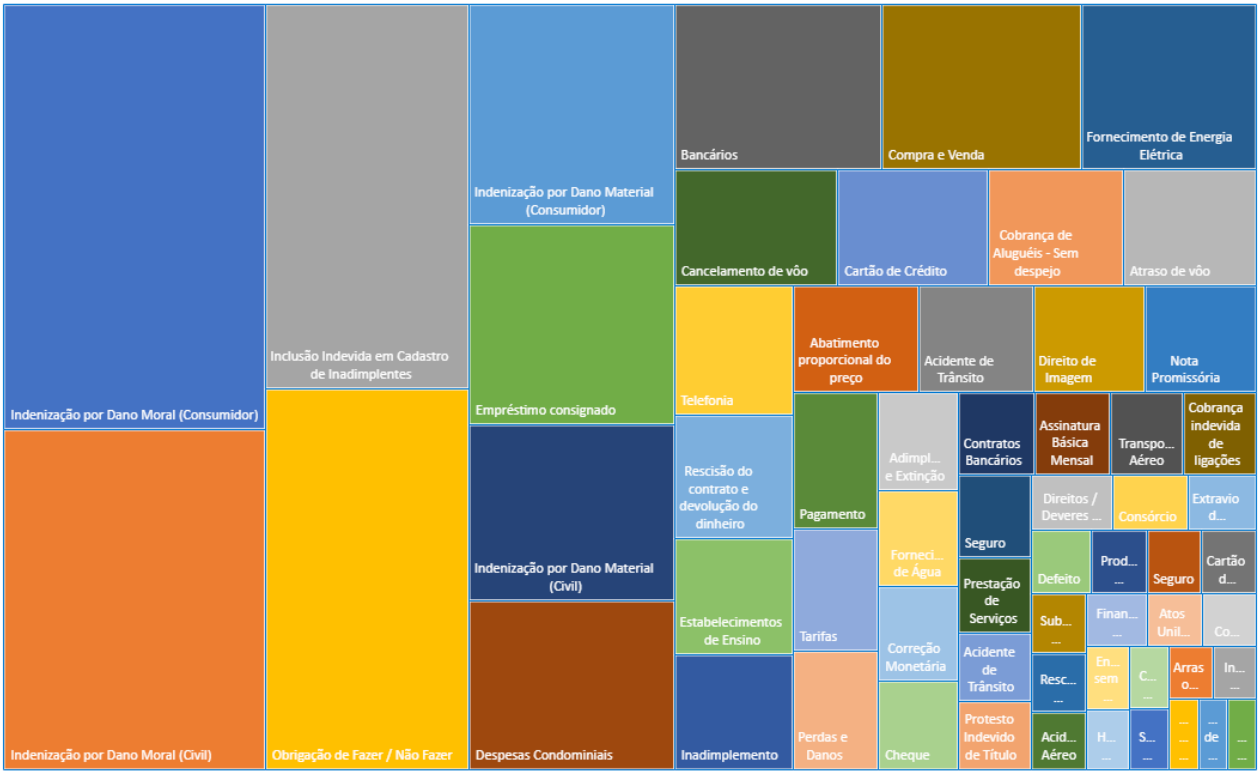
Após aplicação dos filtros, foram identificados 154 assuntos contemplados no quantitativo de 193.358 processos, considerando processos em andamento e arquivados. Vale destacar que mesmo os processos arquivados são levados em consideração; isto porque, mesmo que não estejam em andamento, atenderam a todas as regras dos filtros FA1, FA2, FA3 e FA4.

Parte dos 154 assuntos inicialmente identificados foram aplicados em poucos processos, o que inviabiliza o treinamento para estes assuntos. Com base no quantitativo de 193.358 aplicações de assuntos, considerando um grau de confiança de 95%, com 5% de margem de erro, encontramos o valor mínimo de amostra de 384 exemplos; com isso, foi aplicado o filtro FA5 removendo todos os assuntos que foram aplicados em menos de 384 processos.

Com a aplicação do filtro FA5, restaram 60 assuntos aplicados em 186.664 processos. Estes assuntos passam a compor o grupo de assuntos GA1 cuja seleção considera apenas o quantitativo de processos.

Podemos observar que apenas os filtros FA2 e FA4 são fruto das diretrizes previamente estabelecidas na delimitação do conjunto de dados. Os demais filtros, FA1, FA3 e FA5 são fruto da observação e evolução durante o processo de extração e refinamento dos dados para atender ao propósito de uso em ambiente real. É possível observar ainda que os filtros foram aplicados gradualmente com seus respectivos dados parciais, resultado de uma análise evolutiva que precisa ser feita a cada consulta para que os dados a serem extraídos alcancem o objetivo desejado da melhor forma possível. A Figura 8 ilustra o cenário dos processos por assunto após os filtros.

Figura 8 – Mapa de Árvore depois dos filtros (Processos por assunto).



Fonte: Desenvolvido pelo autor.

Na Figura 8 é possível observar uma maior uniformização nas áreas ocupadas pelos assuntos em comparação com a Figura 5.

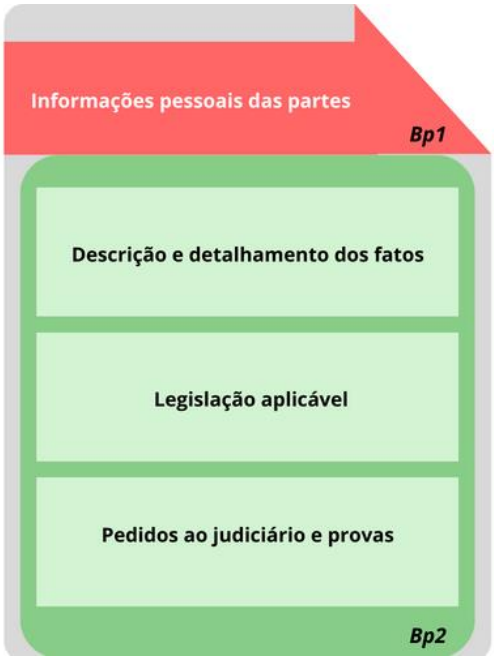
3.2.2. Petições iniciais

As petições iniciais normalmente seguem um padrão de construção contendo, nesta ordem:

1. Informações pessoais das partes;
2. Descrição e detalhamento dos fatos que motivaram a demanda;
3. Legislação aplicável ao caso descrito;
4. Os pedidos ao judiciário e as provas.

Esta é uma abordagem simplificada do padrão de conteúdo da petição inicial definido no Art. 319 da Lei nº 13.105/2015. No item 1 constam as informações dos envolvidos a exemplo de nomes, prenomes, estado civil, profissão, Cadastro de Pessoas Físicas e endereços; no item 2 o detalhamento dos fatos são apresentados para facilitar o entendimento do julgador; no item 3 encontramos os fundamentos jurídicos do pedido, explicitando a legislação relacionada à violação descrita nos fatos; por fim, no item 4 temos os pedidos ao judiciário e as provas relacionadas aos fatos (normalmente enviadas como anexo). Na Figura 9 podemos ver uma ilustração simplificada deste documento:

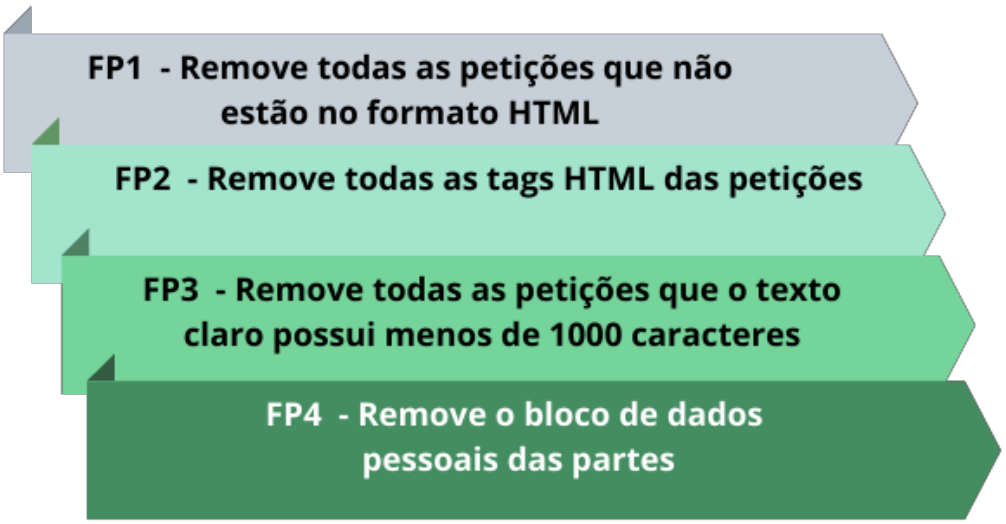
Figura 9 – Representação de uma petição inicial.



Fonte: Desenvolvido pelo autor.

Com base nesse padrão observado ao analisar algumas dezenas de petições iniciais, decidimos separar a petição inicial em dois grandes blocos: Bp1 com as informações do item 1 e Bp2 com as informações dos itens 2, 3 e 4. A extração das petições considerou como base o grupo GA1 de assuntos e aplicou alguns filtros SQL para melhorar a eficiência do modelo a ser gerado posteriormente. A Figura 10 ilustra os filtros aplicados.

Figura 10 – Filtros aplicados nas petições.



Fonte: Desenvolvido pelo autor.

O primeiro filtro FP1 remove todas as petições que não estão em formato HTML, em conformidade com a diretriz número 4. Considerando a necessidade de um maior processamento para arquivos PDFs, armazenamento localizado em serviço de storage separado do banco de dados, a grande quantidade de exemplos em HTML disponíveis para o treinamento e a iniciativa do CNJ em processar todos os documentos dos tribunais em um repositório central que pode ser usado no futuro para treinamento, a decisão por considerar apenas as petições em HTML se mostrou mais eficiente.

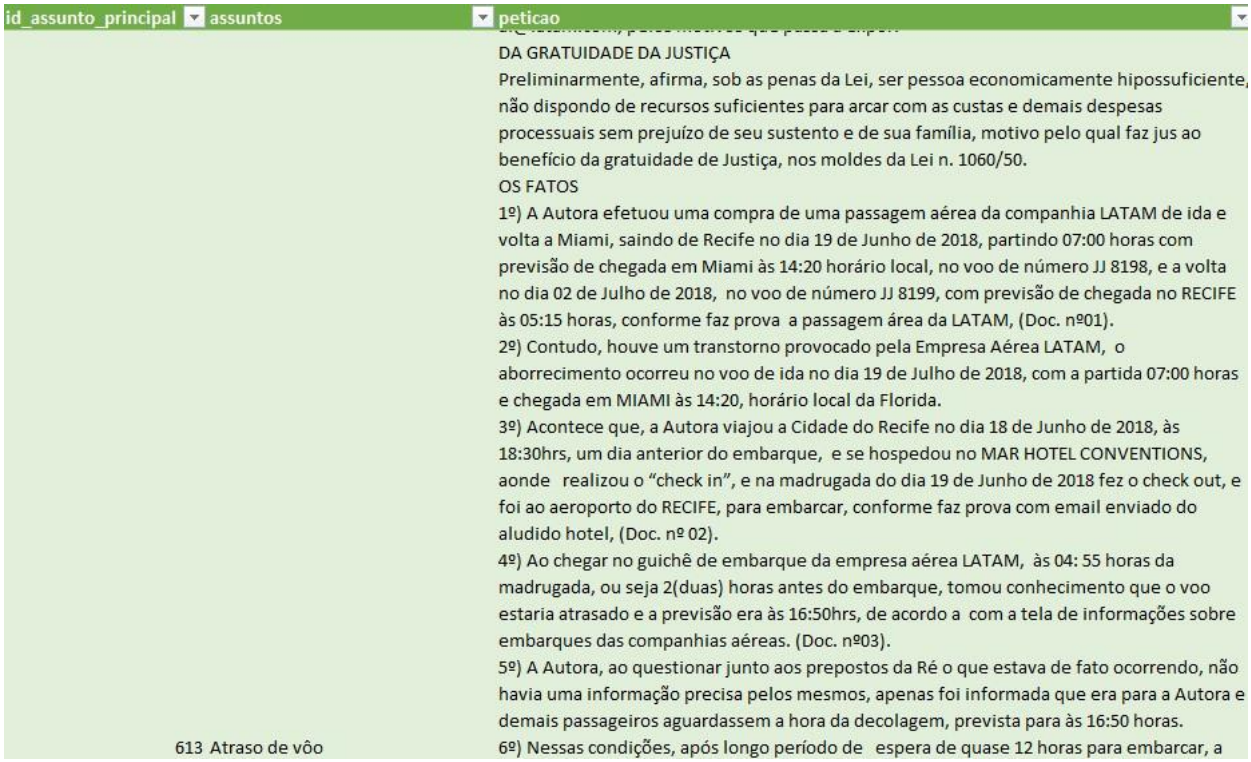
Em seguida, aplicamos o filtro FP2 removendo todas as tags HTML do documento, restando apenas o texto não formatado. A partir disso, aplicamos o filtro FP3 excluindo todas as petições cujo conteúdo, agora apenas texto, possui menos de 1000 caracteres; com isso eliminamos todas as petições consideradas pequenas em conteúdo: notadamente petições cujo conteúdo se resume a “Em anexo”, “Segue em anexo”, “Juntada de petições e documentos” e suas variações.

Por fim, extraímos todas as petições que restaram, juntamente com seus respectivos assuntos e identificador do processo. Apesar dos filtros de remoção de conteúdos aplicados no SQL, as petições selecionadas foram extraídas em sua integralidade para posteriormente serem filtradas com Python, pois a linguagem fornece melhores recursos na interpretação do HTML quando comparada ao SQL do PostgreSQL, o que torna a qualidade da seleção do conteúdo mais precisa.

Para refinar o processo de extração dos dados, utilizamos o módulo BeautifulSoup do Python para interpretar o HTML das petições e extrair apenas o texto puro em conjunto com a biblioteca Pandas para manipulação e seleção dos dados; em seguida aplicamos o filtro FP4 eliminando todo o bloco Bp1, pois além das informações serem sensíveis e protegidas pela Lei Geral de Proteção de Dados, não agregam em nada ao modelo temático proposto. Como resultado, temos o texto bruto do conteúdo semântico das petições abrangendo todo o bloco Bp2, que usamos como entrada para etapa de preparação dos dados.

A Figura 11 ilustra um exemplo dos dados extraídos após aplicação dos filtros e refinamento.

Figura 11 – Exemplo de petição extraída após filtros.



id_assunto_principal	assuntos	peticao
613	Atraso de vôo	DA GRATUIDADE DA JUSTIÇA Preliminarmente, afirma, sob as penas da Lei, ser pessoa economicamente hipossuficiente, não dispondo de recursos suficientes para arcar com as custas e demais despesas processuais sem prejuízo de seu sustento e de sua família, motivo pelo qual faz jus ao benefício da gratuidade de Justiça, nos moldes da Lei n. 1060/50. OS FATOS 1º) A Autora efetuou uma compra de uma passagem aérea da companhia LATAM de ida e volta a Miami, saindo de Recife no dia 19 de Junho de 2018, partindo 07:00 horas com previsão de chegada em Miami às 14:20 horário local, no voo de número JJ 8198, e a volta no dia 02 de Julho de 2018, no voo de número JJ 8199, com previsão de chegada no RECIFE às 05:15 horas, conforme faz prova a passagem área da LATAM, (Doc. nº01). 2º) Contudo, houve um transtorno provocado pela Empresa Aérea LATAM, o aborrecimento ocorreu no voo de ida no dia 19 de Julho de 2018, com a partida 07:00 horas e chegada em MIAMI às 14:20, horário local da Florida. 3º) Acontece que, a Autora viajou a Cidade do Recife no dia 18 de Junho de 2018, às 18:30hrs, um dia anterior do embarque, e se hospedou no MAR HOTEL CONVENTIONS, aonde realizou o "check in", e na madrugada do dia 19 de Junho de 2018 fez o check out, e foi ao aeroporto do RECIFE, para embarcar, conforme faz prova com email enviado do aludido hotel, (Doc. nº 02). 4º) Ao chegar no guichê de embarque da empresa aérea LATAM, às 04: 55 horas da madrugada, ou seja 2(duas) horas antes do embarque, tomou conhecimento que o voo estaria atrasado e a previsão era às 16:50hrs, de acordo a com a tela de informações sobre embarques das companhias aéreas. (Doc. nº03). 5º) A Autora, ao questionar junto aos prepostos da Ré o que estava de fato ocorrendo, não havia uma informação precisa pelos mesmos, apenas foi informada que era para a Autora e demais passageiros aguardassem a hora da decolagem, prevista para às 16:50 horas. 6º) Nessas condições, após longo período de espera de quase 12 horas para embarcar, a

Fonte: Desenvolvido pelo autor.

3.3. Preparação dos dados

Com base nos dados brutos recebidos da etapa anterior para criação do modelo, aplicamos técnicas de mineração para transformar esses dados em um conteúdo que o algoritmo de aprendizado de máquina conseguisse trabalhar com eficiência. Para esta pesquisa, aplicamos as técnicas de remoção de *stopwords*, tokenização, remoção de *tokens* com menos de 2 caracteres e exclusão de *tokens* irrelevantes após análise manual com o auxílio da biblioteca Pandas para manipulação dos dados durante todo o processo.

3.3.1. Limpeza dos dados

Visando facilitar a manipulação dos dados, usamos o recurso da biblioteca Pandas que permite remover as linhas com itens vazios e assim produzir um conjunto de dados mais consistente para o processamento seguinte. Pandas é uma biblioteca em Python que fornece estruturas de dados flexíveis para tornar o trabalho com dados fácil e intuitivo permitindo análises de dados em alto nível de forma prática (PANDAS, 2022).

3.3.2. Definição das *Stopwords*

Para remover as *stopwords* precisamos primeiro construir uma lista de palavras que serão eliminadas do nosso conteúdo, ou seja, usadas como *stopwords*. Para isso a biblioteca NLTK do Python foi escolhida, pois permite fazer o download de *stopwords* padronizadas em diferentes

linguagens. Nesta pesquisa utilizamos o grupo de palavras "*portuguese*" dessa biblioteca como base das *stopwords* utilizadas.

3.3.3. Extensão das *stopwords*

Ainda que a biblioteca NLTK produza um bom conjunto de *stopwords* para utilização nos textos em português, comumente os trabalhos relacionados à mineração de textos possuem uma temática específica. Pensando neste aspecto, uma *wordlist* com palavras frequentemente utilizadas em petições iniciais foi produzida e adicionada em nossa lista de *stopwords* para que fossem removidas por baixo valor discriminativo. Durante o processo de evolução da pesquisa foi possível detectar diversas *stopwords* do contexto jurídico que foram gradualmente adicionadas para compor a lista utilizada, que apesar de não ser exaustiva, possui representatividade suficiente para eliminar discrepâncias percebidas em análises manuais. Não encontramos, na literatura, uma lista de *stopwords* no contexto jurídico. A Figura 12 ilustra as palavras selecionadas para compor a extensão, também disponíveis no apêndice A.

Figura 12 – *Wordlist* adicional para *stopwords*.

'agravo'	'ainda'	'além'	'ante'	'apelação'	'art'	'artigo'	'assim'	'autor'	'autora'
'ação'	'caput'	'caso'	'causa'	'cinco'	'civil'	'cláusulas'	'conforme'	'custas'	'cível'
'código'	'deferido'	'desde'	'desta'	'destas'	'deste'	'destes'	'deve'	'dever'	'dez'
'diante'	'direito'	'dois'	'entendimento'	'então'	'excelência'	'expor'	'exposto'	'fato'	'fatos'
'fim'	'indeferido'	'inicial'	'judicial'	'judiciária'	'juiz'	'julgamento'	'junto'	'juridica'	'justiça'
'juízo'	'legal'	'lei'	'mil'	'nada'	'naquele'	'naqueles'	'neste'	'nestes'	'nestes'
'nove'	'oito'	'ora'	'outra'	'parte'	'parágrafo'	'passa'	'pedido'	'pode'	'pois'
'princípio'	'processo'	'processuais'	'processual'	'promovente'	'promovida'	'promovido'	'prova'	'qualquer'	'quatro'
'razão'	'reais'	'relator'	'requer'	'segue'	'seguinte'	'segundo'	'seis'	'segue'	'ser'
'sete'	'sob'	'sobre'	'tal'	'ter'	'termos'	'tj'	'toda'	'todas'	'todo'
'todos'	'tribunal'	'três'	'tudo'	'um'	'valor'	'vejamos'	'vez'	'vossa'	

Fonte: Desenvolvido pelo autor.

Enquanto a quantidade de *stopwords* padronizadas para a língua portuguesa disponível na biblioteca NLTK possui 207 palavras, a *wordlist* gerada nesta pesquisa analisando os dados das petições iniciais possui 110 palavras. Um incremento de mais de 50% nas *stopwords* utilizadas para tratamento do nosso *corpus*.

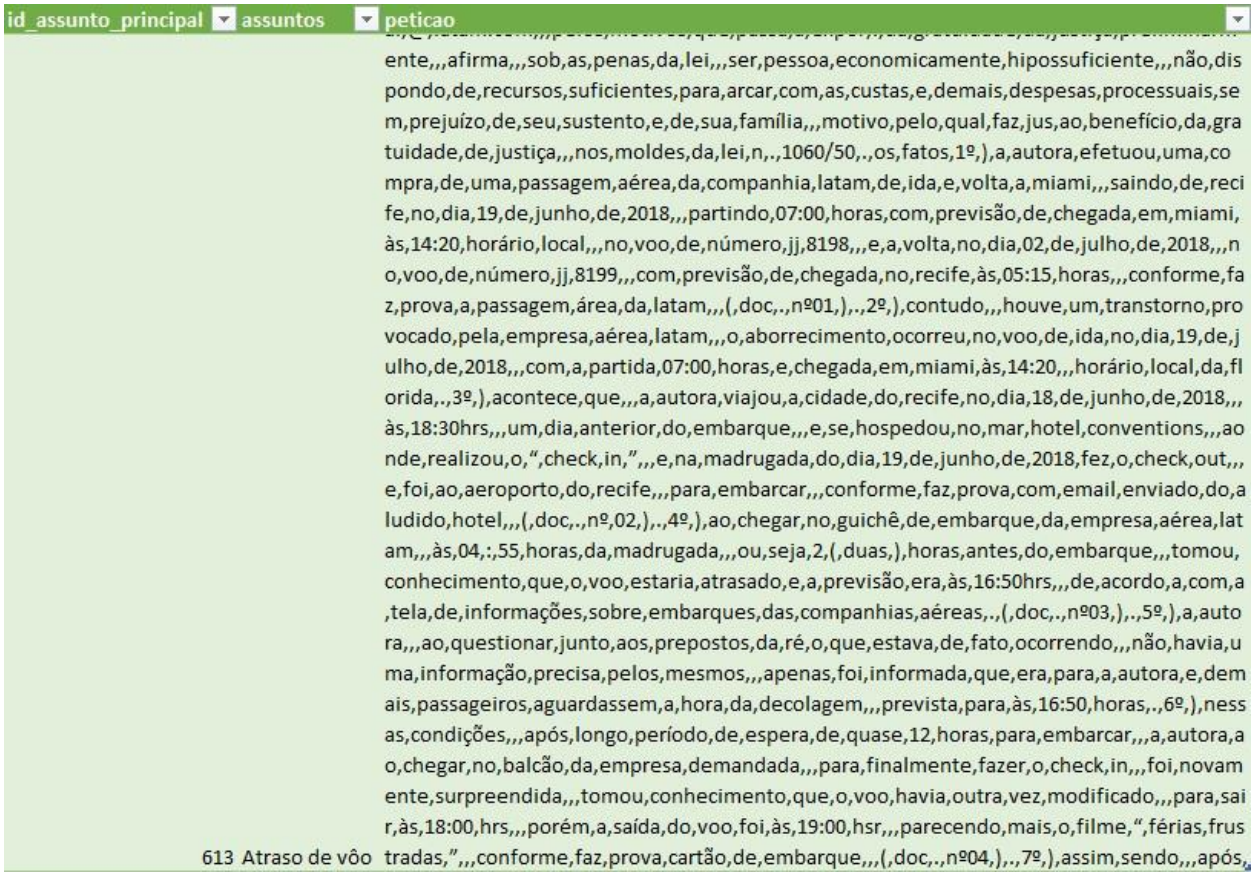
3.3.4. Tokenização

Para execução da técnica *tokenization* utilizamos o recurso da biblioteca NLTK, pois permite a escolha da linguagem em que a lista de *tokens* será gerada com base no texto passado como parâmetro. Em nossa execução, utilizamos a linguagem "*portuguese*".

Após a geração dos *tokens*, convertemos todas as letras em minúsculas visando gerar um conjunto de dados mais uniforme.

A Figura 13 ilustra o resultado da petição ao após tokenização e padronização.

Figura 13 – Petição convertida em *tokens*.



id_assunto_principal	assuntos	peticao
		ente,,,afirma,,,sob,as,penas,da,lei,,,ser,pessoa,economicamente,hipossuficiente,,,não,dispondo,de,recursos,suficientes,para,arcar,com,as,custas,e,demais,despesas,processuais,sem,prejuízo,de,seu,sustento,e,de,sua,família,,,motivo,pelo,qual,faz,jus,ao,benefício,da,gratuidade,de,justiça,,,nos,moldes,da,lei,n,.,1060/50,.,os,fatos,1º,),a,autora,efetuou,uma,compra,de,uma,passagem,aérea,da,companhia,latam,de,ida,e,volta,a,miami,,,saindo,de,recife,no,dia,19,de,junho,de,2018,,,partindo,07:00,horas,com,previsão,de,chegada,em,miami,às,14:20,horário,local,,,no,voo,de,número,jj,8198,,,e,a,volta,no,dia,02,de,julho,de,2018,,,no,voo,de,número,jj,8199,,,com,previsão,de,chegada,no,recife,às,05:15,horas,,,conforme,faz,prova,a,passagem,área,da,latam,,,(,doc,.,nº01,),.,2º,),contudo,,,houve,um,transtorno,provocado,pela,empresa,aérea,latam,,,o,aborrecimento,ocorreu,no,voo,de,ida,no,dia,19,de,julho,de,2018,,,com,a,partida,07:00,horas,e,chegada,em,miami,às,14:20,,,horário,local,da,florida,.,3º,),acontece,que,,,a,autora,viajou,a,cidade,do,recife,no,dia,18,de,junho,de,2018,,,às,18:30hrs,,,um,dia,anterior,do,embarque,,,e,se,hospedou,no,mar,hotel,conventions,,,aonde,realizou,o,“,check,in,”,,,e,na,madrugada,do,dia,19,de,junho,de,2018,fez,o,check,out,,,e,foi,ao,aeroporto,do,recife,,,para,embarcar,,,conforme,faz,prova,com,email,enviado,do,aludido,hotel,,,(,doc,.,nº,02,),.,4º,),ao,chegar,no,guichê,de,embarque,da,empresa,aérea,latam,,,às,04,:,55,horas,da,madrugada,,,ou,seja,2,(,duas,),horas,antes,do,embarque,,,tomou,conhecimento,que,o,voo,estaria,atrasado,e,a,previsão,era,às,16:50hrs,,,de,acordo,a,com,a,tela,de,informações,sobre,embarques,das,companhias,aéreas,.,(,doc,.,nº03,),.,5º,),a,autora,,,ao,questionar,junto,aos,prepostos,da,ré,o,que,estava,de,fato,ocorrendo,,,não,havia,uma,informação,precisa,pelos,mesmos,,,apenas,foi,informada,que,era,para,a,autora,e,demais,passageiros,aguardassem,a,hora,da,decolagem,,,prevista,para,às,16:50,horas,.,6º,),nessas,condições,,,após,longo,período,de,espera,de,quase,12,horas,para,embarcar,,,a,autora,ao,chegar,no,balcão,da,empresa,demandada,,,para,finalmente,fazer,o,check,in,,,foi,novamente,surpreendida,,,tomou,conhecimento,que,o,voo,havia,outra,vez,modificado,,,para,sair,às,18:00,hrs,,,porém,a,saída,do,voo,foi,às,19:00,hsr,,,parecendo,mais,o,filme,“,férias,frustradas,”,,,conforme,faz,prova,cartão,de,embarque,,,(,doc,.,nº04,),.,7º,),assim,sendo,,,após,
613	Atraso de vôo	

Fonte: Desenvolvido pelo autor.

A Figura 14 representa, através de uma nuvem de palavras, os *tokens* mais frequentes de todas as petições envolvidas após a conversão do texto em *tokens*, sem tratamento de *stopwords*.

Figura 14 – Nuvem de palavras com *tokens* antes da remoção das *stopwords*

Fonte: Desenvolvido pelo autor.

3.3.5. Remoção de caracteres especiais e *stopwords*

Com os *tokens* gerados, aplicamos a biblioteca Unidecode para remover os caracteres especiais convertendo o texto dos *tokens* para ASCII, incluindo as *stopwords*.

Com os *tokens* das petições e as *stopwords* sem os caracteres especiais, removemos de cada documento de petição inicial do nosso *corpus* os *tokens* que pertenciam à lista de *stopwords* amplificada com as palavras do contexto jurídico.

3.3.6. Filtragem dos *tokens*

Os *tokens* restantes foram submetidos a um processo de filtragem que remove *tokens* vazios, apenas com dígitos ou menores que 3 caracteres, produzindo o primeiro conjunto de dados apto à execução em algoritmo de aprendizado de máquina.

A Figura 15 ilustra a mudança na nuvem de palavras após a remoção de caracteres especiais, *stopwords* e filtragem dos *tokens*.

Figura 15 – Nuvem de palavras após remoção das *stopwrods*.

Fonte: Desenvolvido pelo autor.

Já na Figura 16 podemos ver o resultado da petição de exemplo após a preparação que cria o *corpus* dessa pesquisa.

Figura 16 – Exemplo de petição preparada.

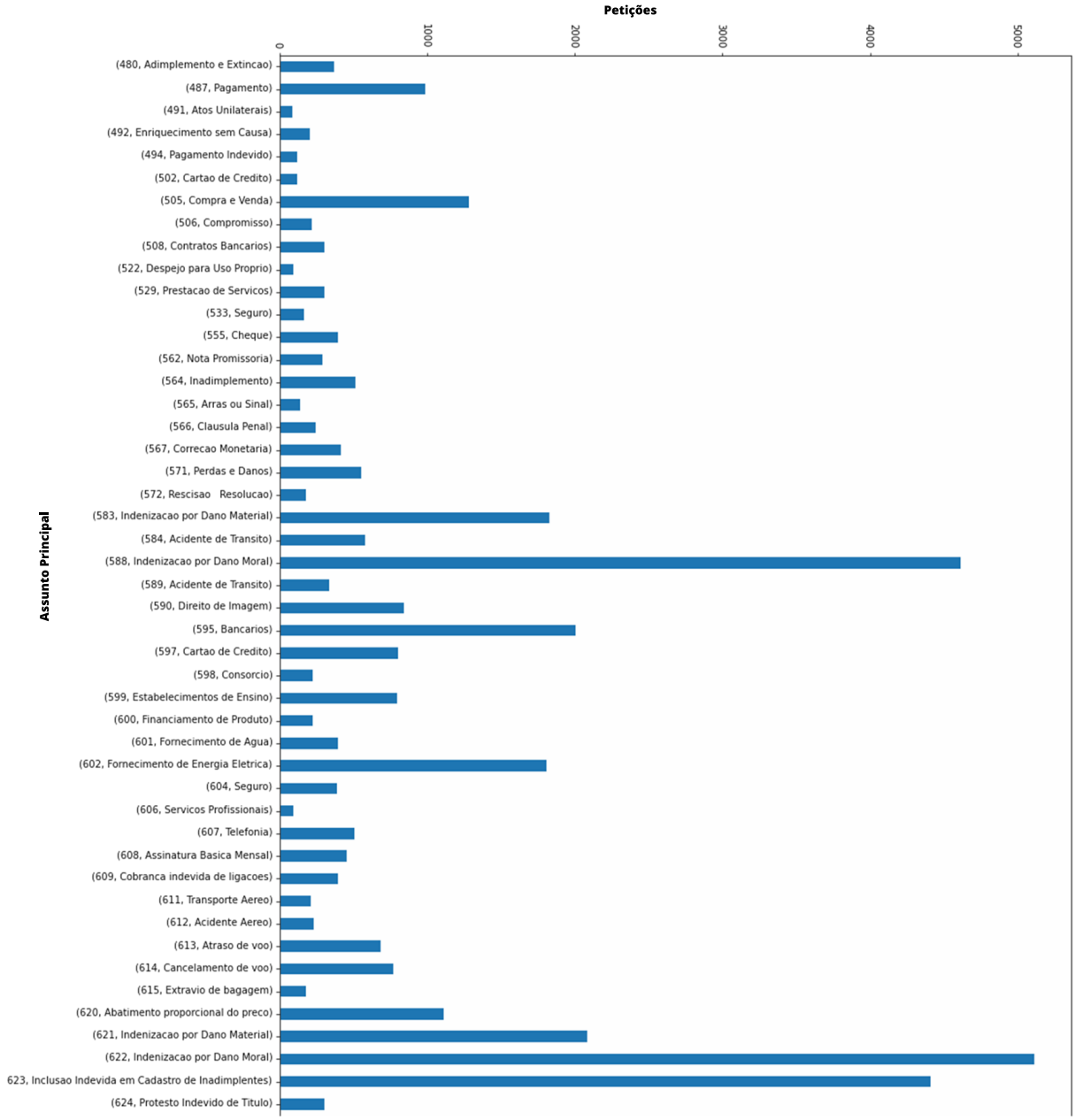
id_assunto_principal	assunto	peticao
		com,motivos,gratuidade,preliminarmente,afirma,penas,pessoa,economicamente,hipossuficiente,dispondo,recursos,suficientes,arcar,custas,demais,despesas,processuais,prejuizo,sustento,familia,motivo,faz,jus,beneficio,gratuidade,moldes,efetuou,compra,passagem,aerea,companhia,latam,ida,volta,miami,saindo,recife,dia,junho,partindo,horas,previsao,chegada,miami,horario,local,voo,numero,volta,dia,julho,voo,numero,previsao,chegada,recife,horas,faz,passagem,area,latam,doc,no01,contudo,transtorno,provocado,empresa,aerea,latam,aborrecimento,ocorreu,voo,ida,dia,julho,partida,horas,chegada,miami,horario,local,florida,acontece,viajou,cidade,recife,dia,junho,18 30hrs,dia,anterior,embarque,hospedou,mar,hotel,conventions,aonde,realizou,check,madrugada,dia,junho,fez,check,out,aeroporto,recife,embarcar,faz,email,enviado,aludido,hotel,doc,chegar,guiche,embarque,empresa,aerea,latam,horas,madrugada,duas,horas,antes,embarque,tomou,conhecimento,voo,estaria,atrasado,previsao,16 50hrs,acordo,tela,informacoes,sobre,embarques,companhias,aereas,doc,no03,questionar,prepostos,ocorrendo,havia,informacao,precisa,mesmos,apenas,informada,demais,passageiros,aguardassem,hora,decolagem,prevista,horas,nessas,condicoes,apos,longo,periodo,espera,quase,horas,embarcar,chegar,balcao,empresa,demandada,finalmente,fazer,check,novamente,surpreendida,tomou,conhecimento,voo,havia,modificado,sair,hrs,porem,saida,voo,hsr,parecendo,filme,ferias,frustradas,faz,cartao,embarque,doc,no04,apos,quartoze,horas,espera,aeronave,decolou,chegando,destino,miami,madruga,horario,local,manha,horario,brasil,previsao,chegada,14 20h,dia,anterior,vale,esclarecer,requerente,escolheu,voo,diurno,poder,chegar,miami,periodo,dia,grande,atraso,causou,aborrecimento,gerando,transtornos,chegar,madrugada,grande,metropoles,miami,dessa,forma,precisou,andar,dentro,aeroporto,miami,maiores,estados,unidos,hora,madrugada,sky train,metro,funcionado,sim,conseguir,pegar,transporte,madrugada,aonde,tudo,dificil,causando,angustia,estresse,desgaste,emocional,apreensao,decepcao,inseguranca,constante,abalo,ordem,moral,material,chegando,exausta,destino,final,horas,manha,vinte,horas,acord
613	Atraso de voo	

Fonte: Desenvolvido pelo autor.

3.4. Descoberta das associações

Ao analisar a quantidade de petições, foi observada uma grande quantidade de exemplos para alguns assuntos, como Indenização por Dano Moral, Indenização por Dano Material, Inclusão Indevida em Cadastro de Inadimplentes, Bancários etc, conforme pode ser visto na Figura 17 e na lista completa no Apêndice D.

Figura 17 – Recorte das petições por assunto.



Fonte: Desenvolvido pelo autor.

Considerando que a característica do algoritmo de associação depende da frequência do item, analisamos e removemos todos os itens que não aparecem mais de 4 vezes no *corpus* de cada assunto, o que representa, aproximadamente, 1% da amostra mínima de petições definida para seleção dos assuntos.

Após a remoção, a quantidade de *tokens* por petição foi analisada conforme Tabela 1, visando identificar a concentração de petições por quantitativo de *tokens*.

Tabela 1 – Quantitativos de *tokens* por petição do *corpus*

Petições	Média	Desvio	Mínimo	1º quartil	Mediana	3º quartil	Máximo
48459	922	667	1	462	811	1234	23914

Fonte: Desenvolvido pelo autor.

Foram selecionados apenas os 1234 *tokens* mais frequentes de cada assunto, considerando o valor encontrado no terceiro quartil, indicando que a maior parte das petições possui até 1234 *tokens*. A partir deste resultado, criamos uma máscara de cada assunto que foi aplicada em cada petição do *corpus* original, de acordo com seu respectivo assunto.

A Figura 18 ilustra parte da máscara gerada por asssunto que foi aplicada nas petições do *corpus*, exemplificando o assunto Adimplemento e Extinção.

Figura 18 – Parte da máscara gerada com base na análise dos quantitativos de *tokens*.

'especificamente', 'citadas', 'condenada', 'algum', 'ocorreu', 'age', 'firmado', 'dor', 'representa', 'empresas', 'periculum', 'proibicao', 'pais', 'obrigatoria', 'referentes', 'pericia', 'cabiveis', 'apresentada', 'concedidos', 'frustrada', 'resumo', 'grau', 'curso', 'inserido', 'jus', 'chamada', 'principal', 'integral', 'causal', 'impende', 'condenado', 'criterio', 'determinado', 'conhecer', 'combinado', 'necessaria', 'exista', 'criminais', 'colendo', 'negocial', 'abrange', 'recebeu', 'presta', 'desequilibrio', 'domicilio', 'ordenar', 'civeis', 'individuais', 'restituido', 'mostra', 'fundamento', 'precedente', 'consideracoes', 'rejeicao', 'admitida', 'economia', 'garantia', 'local', 'averiguar', 'iniquas', 'economico', 'acerca', 'tema', 'calculos', 'oriundos', 'estando', 'decretada', 'lesado', 'abalo', 'atual', 'restou', 'unica', 'acrescida', 'linha', 'coletivos', 'referidas', 'incompativeis', 'celebrados', 'alcance', 'responsabilizado', 'compreensao', 'tac', 'agravada', 'incisos', 'posiciona', 'estabelecidos', 'comparecer', 'corrente', 'altera', 'justo', 'insuficientes', 'residencia', 'alegada', 'negativacao', 'pacificada', 'coloquem', 'postal', 'admite', 'porto', 'salario', 'hoje', 'atendimento', 'informado', 'adequacao', 'manifesta', 'certa', 'obrigue', 'pobre', 'referem', 'deveriam', 'onerosa', 'tendencia', 'sim', 'regulamento', 'diversas', 'dje', 'inaudita', 'testemunhal', 'desatencao', 'obrigou', 'incidencia', 'pessoais', 'observar', 'otica', 'variacao', 'difusos', 'codecon', 'consumidora', 'objetivos',

Fonte: Desenvolvido pelo autor.

A aplicação da máscara removeu todos os *tokens* que não pertenciam aos *tokens* de cada assunto da máscara. Com isso, a quantidade de *tokens* de cada petição diminuiu, produzindo algumas petições com quantidade de tokens irrelevante para análise. A Tabela 2 mostra o resultado após aplicação da máscara.

Tabela 2 – *Tokens* por petição após máscara

Petições	Média	Desvio	Mínimo	1º quartil	Mediana	3º quartil	Máximo
48459	684	472	1	353	613	907	16595

Fonte: Desenvolvido pelo autor.

Com base no primeiro quartil, removemos todas as petições que possuiam menos de 35 *tokens*, aproximadamente 10% do primeiro quartil.

Em seguida, visando uniformizar a quantidade de exemplos de petições por assunto, analisamos a quantidade de petições no *corpus* agrupado por assunto. A Tabela 3 ilustra o resultado.

Tabela 3 – Quantitativo de petições por assunto

Assuntos	Média	Desvio	Mínimo	1º quartil	Mediana	3º quartil	Máximo
60	800	1126	80	186	378	790	5084

Fonte: Desenvolvido pelo autor.

Com base no terceiro quartil, limitamos o número de exemplos por assunto em 790, nivelando a quantidade de exemplos disponíveis para cada assunto, excluindo os exemplos com menor número de *tokens*. Após os ajustes, o *corpus* ficou com as seguintes características a serem usadas pelo algoritmo de associação.

Tabela 4 – Quantitativo de *tokens* por petição após refinamento

Petições	Média	Desvio	Mínimo	1º quartil	Mediana	3º quartil	Máximo
25762	829	560	35	398	753	1140	16595

Fonte: Desenvolvido pelo autor.

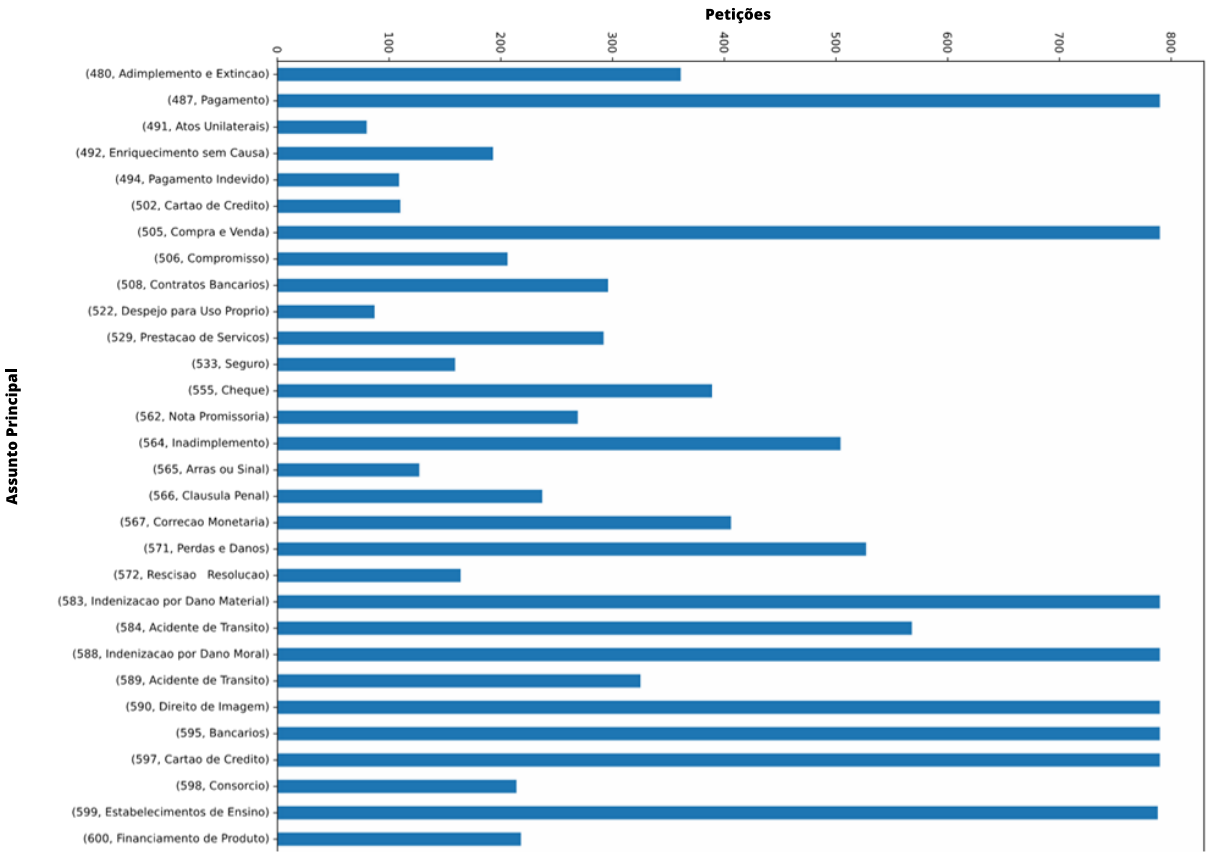
Tabela 5 – Quantitativo de petições após refinamento

Assuntos	Média	Desvio	Mínimo	1º quartil	Mediana	3º quartil	Máximo
60	429	272	80	186	378	788	790

Fonte: Desenvolvido pelo autor.

Na Figura 19, podemos observar um recorte do Apêndice E, após o refinamento que limitou a quantidade de petições de exemplo por assunto, o que garantiu uma uniformidade maior no *corpus* que foi utilizado pelo algoritmo de associação. Esse procedimento foi importante para assegurar a qualidade dos resultados obtidos pelo algoritmo.

Figura 19 – Recorte das petições por assunto após refinamento.



Fonte: Desenvolvido pelo autor.

Para identificar associações entre o assunto e os *tokens* dos textos utilizamos a biblioteca Mlxtend, uma vez que permite a execução do algoritmo de associação *Apriori*. Nos parâmetros de execução definimos o uso do suporte em 50% com tamanho máximo de 3 itens por conjunto de associações.

Submetemos o *corpus* por assunto, separadamente, para identificar dentre os documentos de cada assunto quais seriam os itens frequentes que atendem aos requisitos propostos. Durante o processo de identificação dos itens, percebemos a presença de *tokens* que não possuiam qualquer relação com o assunto analisado. Como resultado, ajustamos o algoritmo para corrigir a discrepância e passamos a verificar a consistência semântica da relação entre o assunto e os itens identificados para cada assunto analisado.

Após a identificação dos itens que compõem as associações de cada assunto, uma máscara contendo todos os itens frequentes do *corpus*, denominada máscara de associação, foi criada e aplicada nos *tokens* de cada petição do *corpus*, removendo os que não pertencem ao conjunto da máscara.

Na Figura 20 é possível observar alguns exemplos do conjunto total de *tokens* identificados de cada assunto, que em sua totalidade contabilizam 1674 *tokens* únicos para compor a máscara de associação.

Figura 20 – Parte dos *tokens* que compõem a máscara de associação

fundamentos, seguir, situacao, encontra, inicialmente, gratuita, forma, poder, acesso, fazer, igualdade, douto, julgador, sabido, assistencia, simples, impossibilidade, despesas, peticao, preceitua, evitar, alguem, busca, defesa, direitos, decorrencia, condicao, social, meios, economicos, prestacao, duas, fundamentais, contrato, financiamento, alienacao, fiduciaria, banco, veiculo, ano, ocorre, mesma, assinou, demandante, outro, totalmente, fica, devidamente, consignado, diferenca, contratuais, empresa, pagou, registro, cidade, copia, receber, financeira, valores, demonstra, anexo, instituicao, contratada, total, centavos, tarifa, cadastro, consideradas, abusivas, serem, onus, servicos, prestados, consumidor, vale, ressaltar, erro, mostra, completamente, documentos, uso, visto, contratos, nenhum, necessidade, boa, juros, tarifas, ilegais, financeiras, portanto, tais, cobrancas, requerida,

Fonte: Desenvolvido pelo autor.

A Figura 21 ilustra os *tokens* de parte de uma petição do assunto Adimplemento e Extinção antes da aplicação da máscara de associação. Neste estágio, a petição tokenizada é composta por 1312 *tokens* em sua totalidade.

Figura 21 – Parte de uma petição tokenizada antes da aplicação da máscara

initio, situacao, encontra, requesta, inicialmente, gratuita, forma, poder, acesso, fazer, valer, igualdade, douto, julgador, sabido, eficacia, assistencia, gratuita, basta, simples, expondo, impossibilidade, constituinte, custear, despesas, proferido, peticao, preceitua, penal, evitar, alguem, frustrada, busca, defesa, direitos, decorrencia, condicao, social, insuficiencia, meios, economicos, resumo, prestacao, assistencia, visa, assegurar, duas, garantias, fundamentais, igualdade, acesso, foro, competente, presente, discute, questoes, mostram, conexao, relacao, consumo, portanto, inicialmente, justificar, escolha, foro, dirimir, questao, apresentada, requerente, invoca, dispositivo, constante, defesa, consumidor, onde, estampa, possibilidade, propositura, domicilio, requerente, eventuais, contratos, tacitos, prestacao, servicos, publicos, consumo, forma, existencia, relacao, consumo,

Fonte: Desenvolvido pelo autor.

Após a aplicação da máscara de associação na petição representada na Figura 21, apenas 518 *tokens* passaram a compor o extrato da petição, conforme pode ser visto na Figura 22. Uma redução de aproximadamente 60% dos *tokens* da petição original.

Figura 22 – Parte de uma petição da após aplicação da máscara de associação

situacao, encontra, inicialmente, gratuita, forma, poder, acesso, fazer, igualdade, douto, julgador, sabido, assistencia, gratuita, simples, expondo, impossibilidade, despesas, peticao, preceitua, evitar, alguem, busca, defesa, direitos, decorrencia, condicao, social, meios, economicos, prestacao, assistencia, duas, fundamentais, igualdade, acesso, presente, conexao, relacao, consumo, portanto, inicialmente, escolha, questao, requerente, dispositivo, defesa, consumidor, onde, possibilidade,

Fonte: Desenvolvido pelo autor.

O resultado extratificado de cada petição do *corpus* com itens que possuem a relevância definida na máscara de associação foi armazenado para ser submetido ao algoritmo de classificação.

3.5. Modelo de classificação

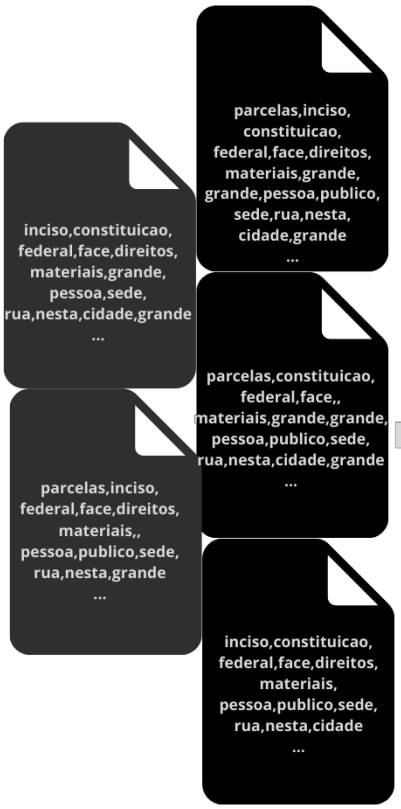
Utilizamos a biblioteca *sklearn* para execução do algoritmo *SVM* utilizando como entrada os dados das petições gerados pelo algoritmo de associação e os dados das petições existentes antes da etapa de aplicação do algoritmo de associação.

Para a execução do algoritmo, foi preciso definir a quantidade de exemplos para ser utilizada como treinamento e teste do modelo. Executamos o algoritmo utilizando a abordagem 70% para treino e 30% para teste, por ser a abordagem comumente utilizada. Rácz, Bajusz e Héberger (2021) compararam combinações de abordagens, entre elas a 70/30 na divisão de treino e teste na classificação multiclasse demonstrando que a abordagem 70/30 e 80/20 atuam melhor em grandes conjuntos de dados como o caso da presente pesquisa.

3.5.1. Vetorização

Para o processo de treinamento do algoritmo, os *tokens* foram transformados em *features* que são as colunas de uma matriz. A matriz gerada possui todos os documentos do *corpus* como linhas que foram analisadas quanto a presença ou não dessas *features*. Esse processo, chamado de vetorização, gera uma matriz numérica ou booleana que é usada para treino ou teste do do algoritmo de classificação. De acordo com Manjunath *et al.* (2021), vetorização é o processo de converter texto em um vetor, podendo ser chamado também de extração de *features*. Tais *features* são a representação das palavras no vetor numérico. Manjunath *et al.* (2021) ainda relatam que diferentes métodos podem ser utilizados para tal processo, mas que a *TF-IDF* (*Term Frequency–Inverse Document Frequency*) é uma técnica que mostra a significancia das palavras de um documento. Nesta pesquisa, optamos por utilizar *TF-IDF* no processo de vetorização. A Figura 23 ilustra o processo de vetorização.

Figura 23 – Ilustração do processo de vetorização



	parcelas	inciso	constituicao	federal	face	direitos	materiais	grande	pessoa	publico	sede	rua	nesta	cidade
doc1	0.2	0.19	0.36	0.53	0.70	0.57	0.104	0.8	0.2	0.2	0.2	0.2	0.2	0.2
doc2	0.23	0.20	0.17	0.14	0.11	0.4	0.5	0.2	0.1	0.4	0.7	0.10	0.13	0.16
dpc3	0.44	0.21	0.2	0.25	0.48	0.71	0.34	0.117	0.140	0.163	0.186	0.209	0.232	0.255
doc4	0.65	0.22	0.21	0.64	0.107	0.150	0.193	0.236	0.279	0.322	0.365	0.408	0.451	0.494
doc5	0.86	0.23	0.40	0.103	0.166	0.229	0.292	0.355	0.418	0.481	0.544	0.607	0.670	0.733

Fonte: Desenvolvido pelo autor.

3.5.2. Modelo *Apriori/SVM*

O conjunto de treino do modelo *Apriori/SVM* restou composto por 18.029 documentos randomizados entre os assuntos, enquanto o conjunto de teste ficou composto por 7.728 documentos. O conjunto de treino foi submetido ao processo de vetorização para identificação das *features* e criação da matriz, resultando em 3425 *features*. A matriz criada foi submetida ao algoritmo de classificação *SVM*, em que a presença ou ausência de determinadas *features* e seus respectivos pesos determinam qual o assunto aquele documento pertence.

3.5.3. Modelo *SVM*

Um segundo *corpus* tokenizado, com *stopwords* removidas, mas sem aplicação do algoritmo *Apriori,* foi submetido ao processo de vetorização e posteriormente ao algoritmo de classificação *SVM*. Neste *corpus* o conjunto de treino foi composto por 33.915 documentos enquanto o conjunto de teste foi composto por 14.536 documentos, gerando 5000 *features*. A matriz gerada foi submetida ao algoritmo de classificação *SVM* nas mesmas configurações executadas para o modelo *Apriori/SVM*.

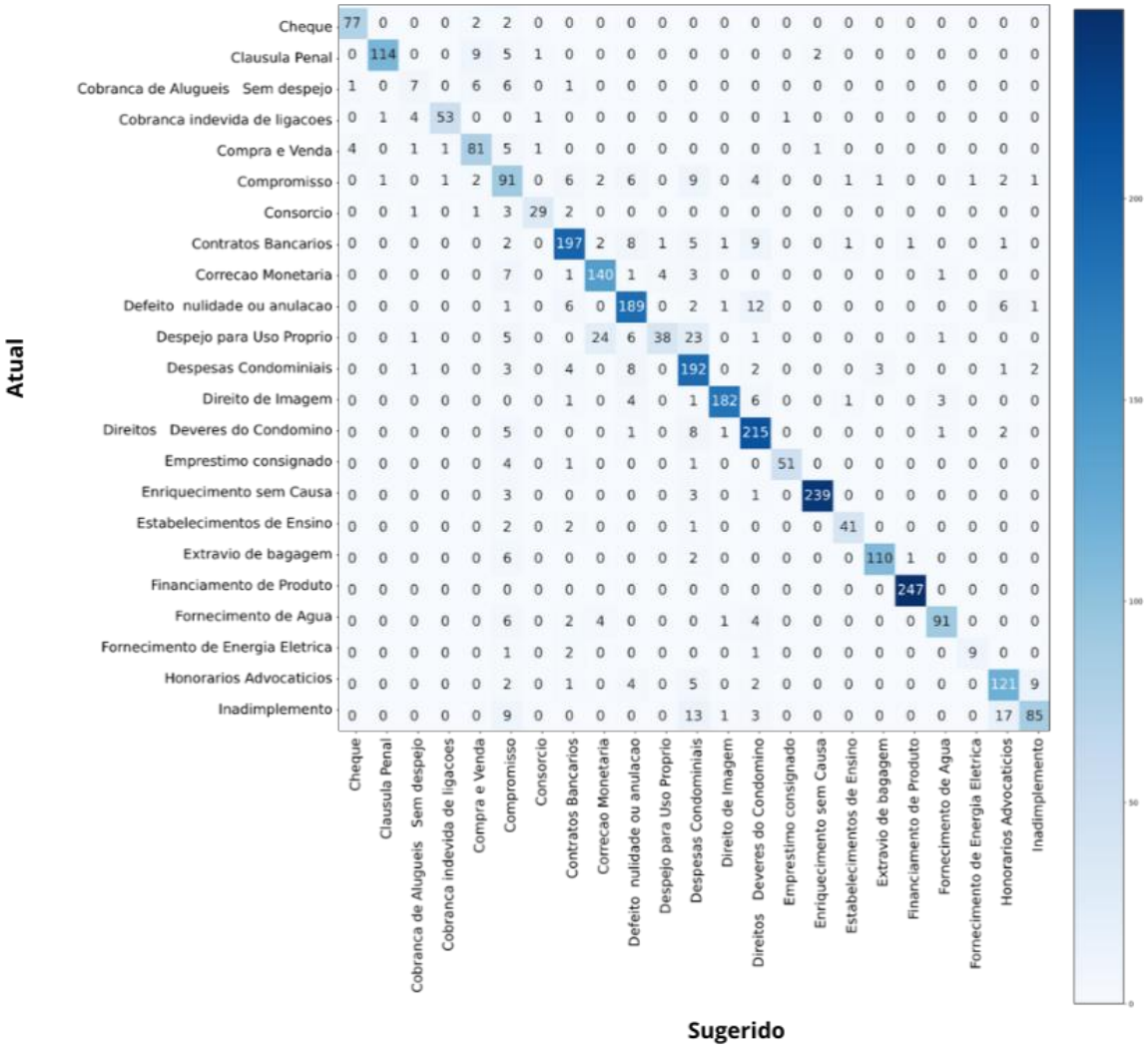
4. AVALIAÇÃO DO MODELO

Os modelos gerados foram avaliados isoladamente para verificação de sua performance de acurácia e *F1 Score*. Este capítulo apresenta as avaliações estatísticas realizadas nos modelos produzidos.

4.1. Avaliação isolada

O modelo *Apriori/SVM* gerado alcançou 77,94% de acurácia e 77,14% de *F1 Score*. Um recorte da matriz de confusão é representado pela Figura 24 ilustrando o desempenho do modelo. Nesta Figura, o eixo vertical representa a classe atual, considerada correta como premissa para o treinamento do modelo, enquanto o eixo horizontal representa a classe prevista pelo modelo. Por exemplo, na linha da classe "Cheque", dos 81 exemplos existentes, 77 foram classificados corretamente pelo modelo e quatro incorretamente. Já na coluna "Cheque", o modelo classificou 82 exemplos como pertencentes a essa classe, mas 4 deles eram, na verdade, da classe "Compra e Venda" e 1 da classe "Cobrança de Aluguéis Sem Despejo".

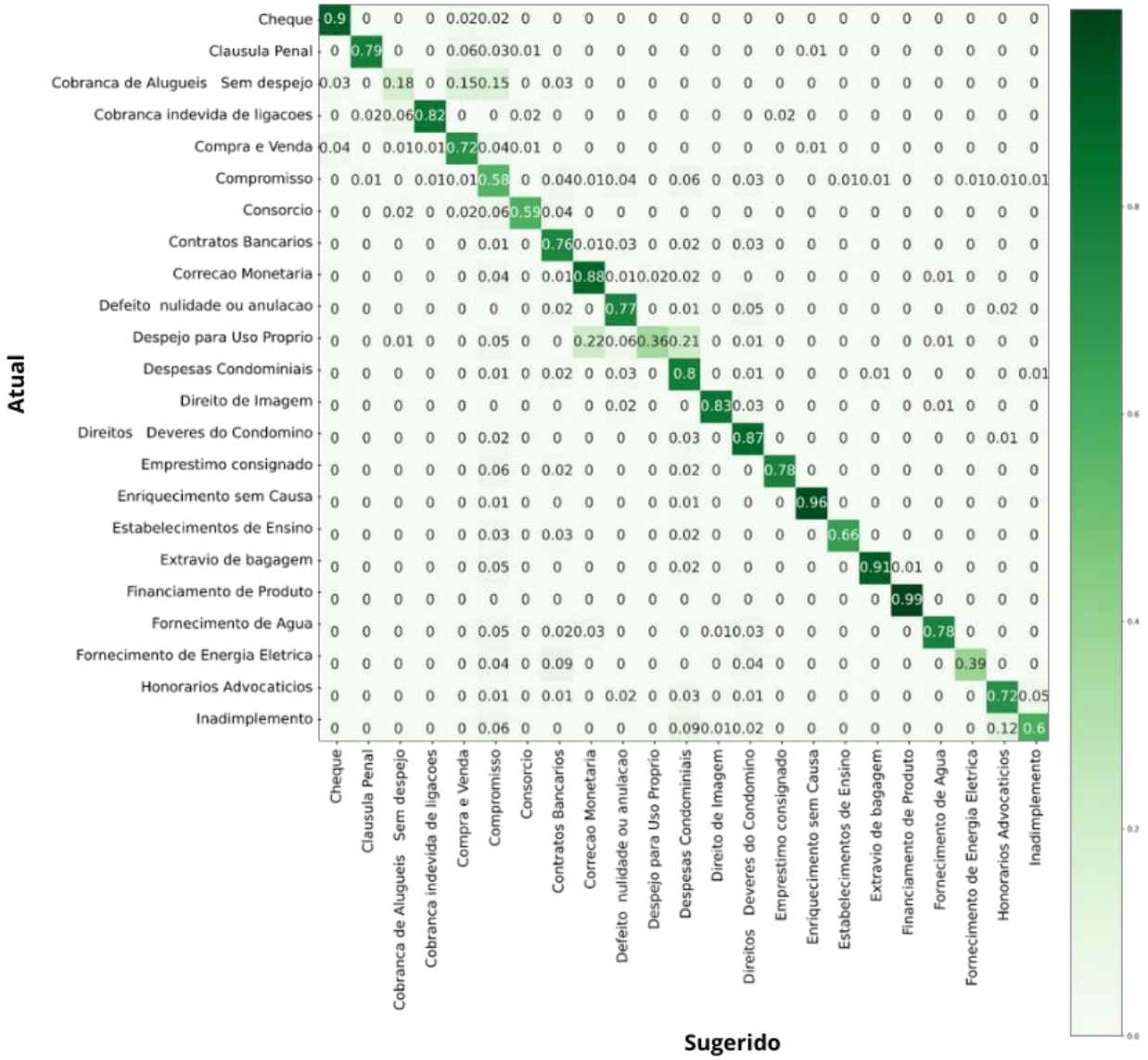
Figura 24 – Recorte da matriz de confusão da classificação *Apriori/SVM*.



Fonte: Desenvolvido pelo autor.

Também foi gerada uma matriz de confusão normalizada, cujo recorte pode ser visto na figura 25.

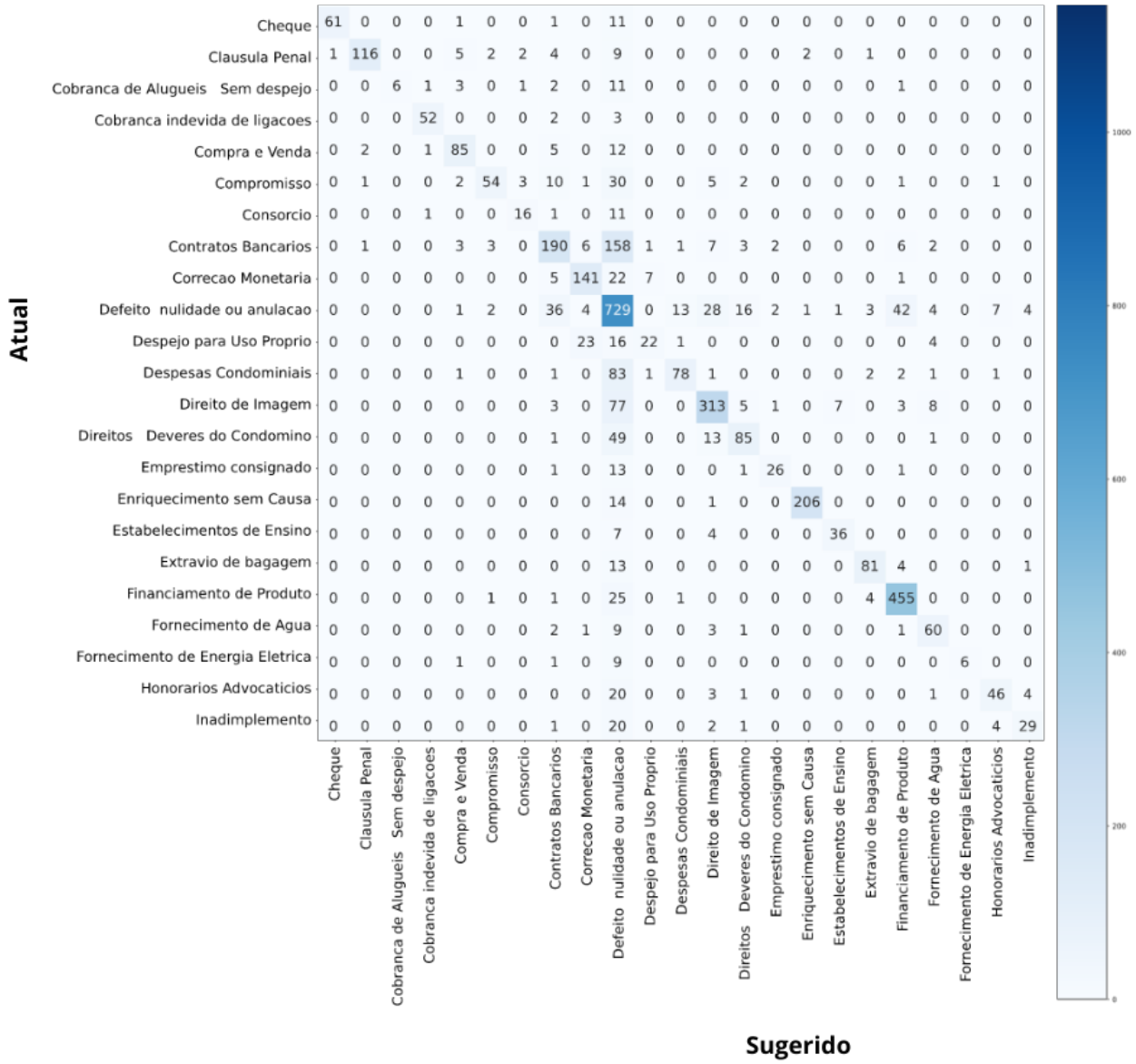
Figura 25 – Recorte da matriz de confusão normalizada *Apriori/SVM*.



Fonte: Desenvolvido pelo autor.

O processo de treinamento do algoritmo de classificação neste segundo *corpus* demorou 6 vezes mais em comparação com o primeiro *corpus*, resultando num modelo com 61,45% de acurácia e 60,65% de *F1 Score*. A Figura 26 representa o recorte da matriz de confusão dessa segunda classificação.

Figura 26 – Recorte da matriz de confusão da classificação *SVM*.

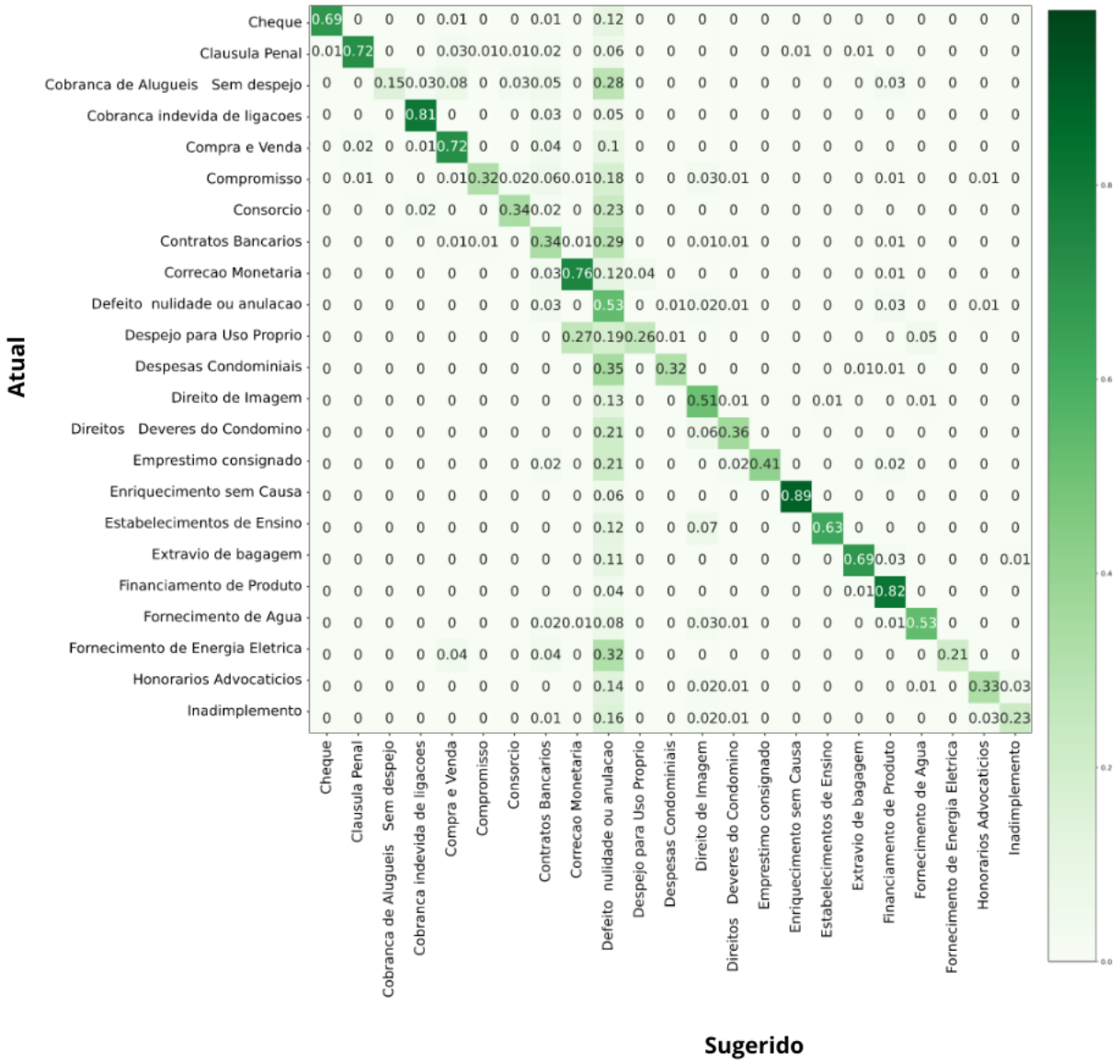


Fonte: Desenvolvido pelo autor.

Apesar da grande quantidade de linhas e colunas da matriz, é possível perceber visualmente a redução das linhas/colunas em destaque na Figura 26 quando comparada à Figura 24 que representa a matriz de confusão da aplicação *Apriori/SVM*. Quanto maior o número de linhas/colunas em destaque na diagonal da matriz, maior a assertividade do modelo, pois indica que o rótulo indicado no conjunto de testes é o mesmo sugerido pelo modelo de classificação ao analisar tal conjunto.

Um recorte da matriz de confusão normalizada da classificação utilizando apenas *SVM* está representado na Figura 27.

Figura 27 – Matriz de confusão normalizada *SVM*



Fonte: Desenvolvido pelo autor.

Na matriz normalizada da Figura 27 é possível notar uma grande quantidade de linhas/colunas se destacando espalhadas na matriz, indicando que apesar das linhas/colunas em destaque na diagonal representando os acertos, há uma grande quantidade de erros também.

Tabela 6 – Comparação das avaliações das execuções

	SVM	*Apriori /SVM*	Diferença nominal	Diferença percentual
Acurácia	61,45%	77,94%	+16,49%	+26,83%
F1 Score	60,65%	77,35%	+16,70%	+27,54%

A Tabela 6 demonstra o comparativo das duas execuções apontando um aumento de 26,83% na acurácia do modelo *Apriori/SVM* quando comparado com o modelo apenas *SVM* e um aumento de 27,54% no *F1 Score* na mesma comparação.

4.2. *10-fold Cross-Validation*

O modelo gerado na abordagem *Apriori/SVM*, apesar da melhoria significativa quando comparada ao modelo apenas *SVM*, foi comparado em uma execução isolada, sendo necessária a execução de várias iterações utilizando diferentes partes dos dados embaralhados para validação da performance do modelo e uma maior segurança na aplicação desse modelo em ambiente produtivo.

Para validar os resultados obtidos de uma execução isolada, aplicamos a técnica de validação cruzada *10-fold Cross-Validation*. Para divisão do *corpus*, utilizamos a classe *StratifiedKold* dividindo o *corpus* em 10 subconjuntos com a mesma quantidade de amostras por assunto judicial. A técnica foi aplicada tanto no *corpus* com aplicação do *Apriori* quanto no *corpus* sem aplicação do *Apriori* calculando as métricas de acurácia, precisão, revocação e *F1 Score*. Os resultados das execuções podem ser vistos na Tabela 7.

Tabela 7 – *10-Fold Cross-Validation*

Acurácia		Precisão		Revocação		*F1 Score*	
SVM	***Apriori/SVM***	***SVM***	***Apriori/SVM***	***SVM***	***Apriori/SVM***	***SVM***	***Apriori/SVM***
62,26%	79,19%	71,67%	80,46%	50,13%	68,91%	55,51%	71,87%
64,29%	80,71%	70,78%	83,62%	52,88%	72,44%	57,61%	75,44%
62,87%	80,59%	63,30%	81,55%	52,14%	71,63%	55,34%	74,48%
63,86%	80,59%	73,60%	78,80%	53,72%	70,30%	58,56%	72,46%
62,62%	79,85%	71,25%	79,59%	52,34%	69,26%	56,75%	72,09%
63,03%	80,78%	65,50%	80,93%	50,84%	71,56%	54,91%	73,86%
63,12%	79,66%	64,52%	80,87%	50,22%	68,99%	54,36%	71,57%
63,41%	80,54%	69,86%	82,28%	51,87%	71,81%	55,93%	74,51%
63,01%	79,15%	70,30%	78,50%	51,56%	68,56%	55,89%	71,45%
62,15%	80,00%	67,56%	79,89%	52,10%	70,85%	56,59%	73,06%

Em todas as execuções, há uma melhoria considerável quando aplicamos o modelo *Apriori/SVM*. Com base no *F1 Score*, é possível identificar que a combinação do algoritmo de associação *Apriori* com o algoritmo de classificação *SVM*, quando comparada com a execução apenas do *SVM*, resulta em uma melhoria na classificação do assunto judicial em mais de 15%. A Tabela 8 ilustra a melhoria alcançada nas médias das métricas observadas.

Tabela 8 – Média das métricas alcançadas pós *10-fold Cross-Validation*

	SVM	*Apriori /SVM*	Diferença nominal	Diferença percentual
Acurácia	63,06%	80,11%	17,05%	27,04%
Precisão	68,83%	80,65%	11,82%	17,17%
Revocação	51,78%	70,43%	18,65%	36,02%
F1 Score	56,15%	73,08%	16,93%	30,15%

A Tabela 8 apresenta uma melhoria ainda maior que a execução isolada apresentada na Tabela 6, tanto na diferença nominal quanto na diferença percentual, indicando uma performance promissora da abordagem *Apriori/SVM* na mineração de textos mesmo quando submetida à validação cruzada com 10 execuções. Destaque para a revocação com a maior melhoria percentual, 36,02%, além das melhorias de 30,15% no *F1 Score* e 27,04% na acurácia.

Quando observamos apenas a acurácia dos dois modelos, nos deparamos com uma melhoria nominal de 17,05% na média das 10 acurácias obtidas na execução da validação cruzada apresentada na Tabela 7 ao aplicar a abordagem *Apriori/SVM*, representando uma melhoria significativa para um modelo de classificação de aprendizado de máquina. Esta melhoria é confirmada ao compararmos a diferença nominal das execuções para a métrica do *F1 Score*, pois a diferença nominal obtida é de 16,93%, muito próxima da melhoria obtida na acurácia das execuções.

4.3. Performance Computacional

Realizamos um teste utilizando 10 petições aleatórias para avaliar o tempo de processamento necessário para o modelo gerado processar o texto bruto, em HTML, da forma como é recebido do editor.

Tabela 9 – Medições de performance das execuções

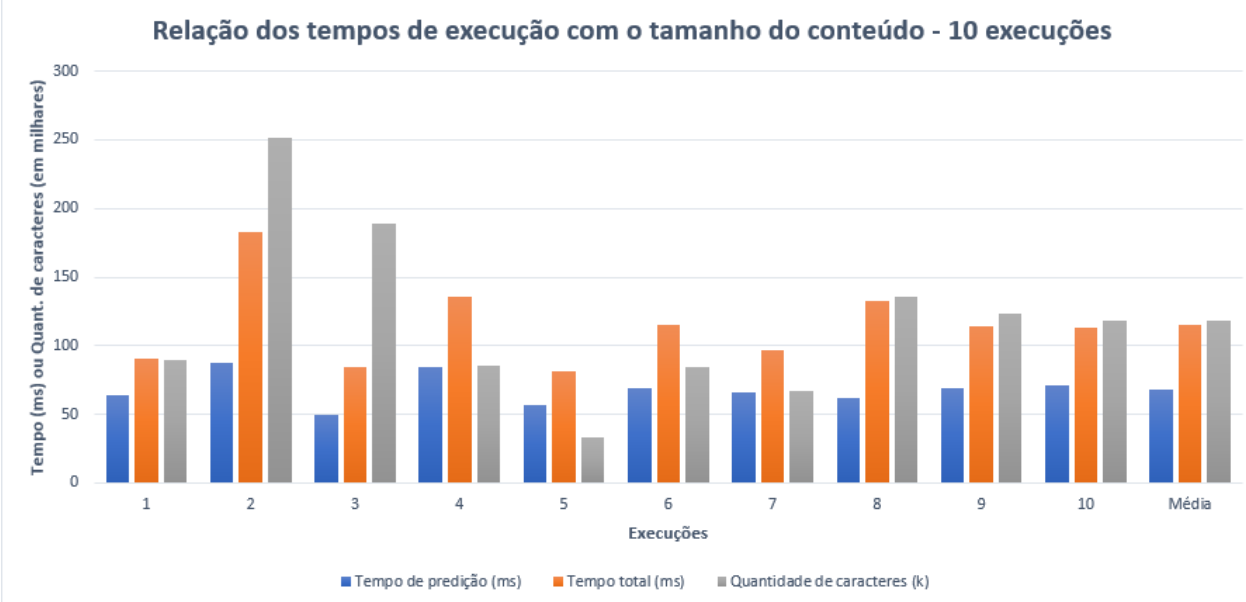
Execução	Tempo de predição (ms)	Tempo total (ms)	Quantidade de caracteres (k)
1	63,923	90,811	89,764
2	87,724	182,728	251,224
3	49,588	84,686	189,308
4	84,772	135,814	85,192
5	57,034	81,273	33,58
6	68,492	115,118	84,584
7	66,1	97,014	67,26
8	61,512	132,825	135,504
9	68,676	114,237	123,32
10	70,897	113,488	117,8
Média	67,8718	114,7994	117,7536

A Tabela 9 apresenta as seguintes informações: a) o tempo de predição em milissegundos; b) o tempo total de processamento, que inclui, além do tempo destinado para a predição, as etapas de limpeza do texto bruto, tokenização, remoção de stopwords e vetorização, em milissegundos; e c) a quantidade de caracteres, em milhares, do texto em HTML.

Observamos que a menor petição avaliada possui 33.580 caracteres e a maior possui 251.224 caracteres, devido à quantidade de tags envolvidas na formatação do texto. Além disso, verificamos que o tempo de predição médio foi de 67,8718 milissegundos. É importante destacar que houve pouca variação no tempo de predição em relação à quantidade de caracteres processados, o que indica uma boa performance do modelo em termos de eficiência.

Também é importante mencionar que, ao avaliarmos a quantidade de caracteres das petições, foi possível perceber uma grande diferença entre a menor e a maior petição. Isso pode ter impacto no tempo de processamento, já que a quantidade de caracteres influencia diretamente na complexidade do processamento. No entanto, mesmo diante dessa variação, o modelo apresentou um tempo de predição bastante estável, o que é um sinal de sua capacidade de lidar com essas diferenças de forma consistente.

Figura 28 – Relação de tempo com quantidade de caracteres



Fonte: Desenvolvido pelo autor.

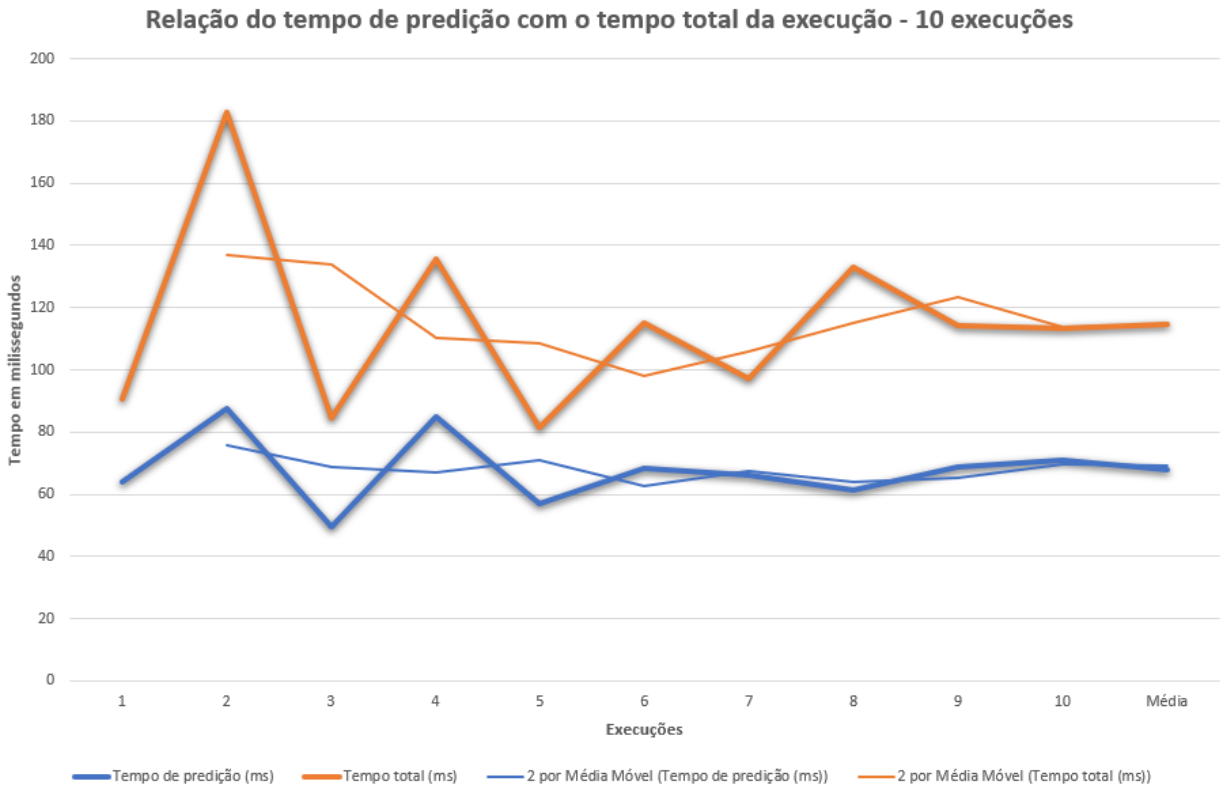
A Figura 28 apresenta um gráfico em barras que ilustra a relação entre o tempo total, o tempo de predição e a quantidade de caracteres envolvidos. É possível observar que há uma correlação entre a quantidade de caracteres e o tempo total de processamento, ou seja, quanto maior a quantidade de caracteres, maior o tempo de processamento. No entanto, é possível perceber também que há um tempo mínimo de processamento, independentemente da quantidade de caracteres. Isso pode ser visto na execução 5, em que a quantidade de caracteres, apesar de menor, não foi suficiente para diminuir o tempo total de processamento.

Esse gráfico é útil para entendermos como o tempo de processamento é influenciado pelo tamanho do texto e também para avaliarmos a performance do modelo. É importante lembrar que o tempo de processamento é um fator importante em aplicações práticas, portanto, é fundamental que o modelo seja capaz de processar os textos de forma rápida e eficiente.

Outro ponto a ser observado nesse gráfico é o tempo de predição em si. Embora a quantidade de caracteres tenha influência no tempo total de processamento, o tempo de predição em si parece estar relativamente constante, independentemente da quantidade de caracteres. Isso pode indicar que o modelo é capaz de realizar a predição de forma rápida, independentemente do tamanho do texto.

Já a Figura 29 apresenta um gráfico que ilustra a variação dos tempos de predição e totais das 10 execuções realizadas. Além disso, o gráfico apresenta uma média móvel de 2 períodos das execuções, o que permite observar uma tendência geral tanto nos tempos de predição quanto nos tempos totais.

Figura 29 – Relação do tempo de predição com o tempo total da execução



Fonte: Desenvolvido pelo autor.

É possível observar que os tempos de predição são relativamente estáveis em comparação ao tempo total da execução, que apresenta mais variação. Contudo, ao compararmos os valores da média móvel com as médias gerais apresentadas na Tabela 9, percebemos que eles se mantêm alinhados, o que indica que a média móvel é uma representação confiável da tendência geral da maior parte das execuções.

5. CONCLUSÃO

A aplicação de técnicas de mineração de textos em um contexto jurídico tem sido cada vez mais explorada devido às suas peculiaridades semânticas e formais contidas nos textos que compõem o *corpus*. Nesta pesquisa, criamos um modelo de inteligência artificial para classificar assuntos judiciais com base nos textos das petições iniciais analisadas. Para isso, utilizamos uma abordagem empírica na seleção dos dados submetidos na criação do modelo juntamente com uma abordagem integrada de algoritmo de associação na seleção das *features* com um algoritmo de classificação.

A combinação de algoritmos demonstrou capacidade promissora de melhorar a performance do modelo criado, selecionando *features* relevantes com base na associação de 1 a 3 *tokens*. Importante ressaltar ainda a relevância dos ajustes feitos no conjunto de dados para otimizar a aplicação do algoritmo de associação, tendo em vista o alto uso de recursos que cada execução demanda.

Desde a extração até o modelo, diversos filtros foram aplicados com o objetivo de eliminar o ruído dos textos jurídicos utilizados para o treinamento e validação do modelo. Tais filtros representam uma forma funcional de selecionar os dados jurídicos em petições que podem ser utilizados para compor o *corpus* do modelo de aprendizado de máquina, levando em consideração os aspectos legais e funcionais necessários à realidade dos sistemas existentes no judiciário, compatíveis com as diretrizes impostas pelo Conselho Nacional de Justiça.

O algoritmo de associação *Apriori* foi utilizado para selecionar as *features* mais relevantes de cada assunto judicial analisado, dividindo os documentos do *corpus* para aplicação individual do algoritmo. Uma vez selecionadas as *features*, uma máscara foi criada para eliminar os demais *tokens* não contidos na máscara daquele assunto. A partir disso o modelo de classificação com algoritmo *SVM* foi treinado na proporção 70% para treino, 30% para teste e posteriormente avaliada a acurácia e o *F1 Score*, atingindo os valores de 77,94% e 77,35% no modelo *Apriori/SVM*.

Uma segundo *corpus*, sem a etapa de aplicação do algoritmo de associação, foi submetido a criação de um modelo com o algoritmo de classificação *SVM* nas mesmas configurações do primeiro modelo. A avaliação apontou uma acurácia de 61,45% e um *F1 Score* de 60,65%, significativamente menor que a abordagem com *Apriori*.

Posteriormente, o modelo *Apriori/SVM* foi validado utilizando a técnica *K-fold Cross-Validation* com o valor de K igual a 10. A validação do modelo *Apriori/SVM* alcançou média 80,11% de acurácia e 73,08% de *F1 Score*, compatível com a performance da execução isolada. Também foi executada a validação do modelo apenas *SVM*, evidenciando, quando comparada cada execução, consistência na melhoria apresentada na Tabela 6, tendo alcançado 36% de melhoria

em uma das métricas analisadas na validação, evidenciando melhoria de performance na execução do algoritmo de classificação quando associada ao algoritmo de associação.

Especialmente no modelo de classificação de assuntos judiciais, uma performance acima de 90% deve ser analisada com bastante cautela, pois os exemplos utilizados pela extração da base de dados podem ter sido errôneamente rotulados, uma vez que a escolha do assunto judicial é discricionária ao demandante, que nem sempre possui o conhecimento necessário para seleção do assunto judicial correto. Por esse motivo, uma performance mais próxima dos 80% pode ser avaliada como uma performance realista e aceitável, pois o modelo tende a acertar sistematicamente na tentativa de corrigir os equívocos na seleção de assunto judicial pelo usuário, que também compoe o *corpus* analisado. O modelo *Apriori/SVM*, integrando algoritmo de associação com algoritmo de classificação alcançou média de 80,11% de acurácia e 73,08% de *F1 Score*, valores muito próximos do que se espera de um modelo aceitável para ser utilizado no ambiente real. Com isso concluímos que a integração do algoritmo de associação com algoritmo de classificação para identificação de assunto judicial, da forma como foi abordada, pode representar um caminho a ser seguido para implantação de modelos com problemática similar no judiciário, assim como uma abordagem a ser validada em outros contextos de mineração e classificação de texto.

5.1. Trabalhos futuros

Com base nos resultados desta pesquisa, a integração do modelo pode ser implantada no PJe, permitindo que sugestões de assuntos com base nos textos das petições possam ser enviadas aos usuários no momento do protocolo do processo. Posteriormente, o mesmo modelo pode ser utilizado para disponibilização de um canal com o cidadão de forma mais acessível, sem tantas barreiras formais, através de um aplicativo ou sistema WEB que permita o cadastro e envio dos dados básicos do ocorrido para a criação da demanda judicial.

O *corpus* utilizado não passou por uma avaliação formal de um comitê com conhecimento das Tabelas Processuais Unificadas. A criação de um grupo ou comitê que possa aplicar uma curadoria nos dados que serão utilizados na criação do modelo pode resultar em um classificador com maior assertividade e qualidade nos assuntos sugeridos.

O CNJ tem promovido o DATAJUD, Base Nacional do Poder Judiciário, responsável pelo armazenamento centralizado dos dados e metatados processuais relativos a todos os processos físicos e eletrônicos dos tribunais. A Portaria 160/2020 do Conselho Nacional de Justiça tornou público cronograma para correção dos dados do DataJud, tendo em vista o grande número de informações enviadas que contrariam as regras definidas para seleção dos dados das Tabelas Processuais Unificadas, entre outras informações processuais. Neste sentido, um modelo de análise e sugestão de assunto judicial pode ser criado expandindo o escopo desta pesquisa para sugerir

quais os assuntos judiciais seriam adequados para correção dos dados apontados pelo DataJud, evitando que a análise textual manual seja feita para reclassificar o processo da forma correta.

A integração do algoritmo de associação *Apriori* com o algoritmo de classificação *SVM* utilizada nesta pesquisa apresentou resultados promissores, podendo ser aprofundada e aplicada em outros contextos que envolvam classificação de textos com *corpus* de volume significativo, aproveitando o benefício da seleção de *features* para redução da quantidade de dados a serem analisados, evitando a perda de textos relevantes para o domínio envolvido. O uso da integração também demonstrou significativa melhoria na performance do modelo, passando a ser uma alternativa na construção de modelos supervisionados de classificação de textos.

Por fim, processos digitalizados cujo assunto judicial definido à época de sua criação não é compatível ou é inexistente nas Tabelas Processuais Unificadas podem ter suas petições submetidas à análise de um modelo expandido para reclassificação destes processos de forma automática.

REFERÊNCIAS BIBLIOGRÁFICAS

AGUIAR, A. *et al.* Text classification in legal documents extracted from lawsuits in brazilian courts. **Brazilian Conference on Intelligent Systems**, 2021. p. 586-600.

AGUIAR, A. *et al.* Using Topic Modeling in Classification of Brazilian Lawsuits. **International Conference on Computational Processing of the Portuguese Language.**, 2022. Springer, Cham. p. 233-242.

ALBON, C. **Machine learning with python cookbook:** Practical solutions from preprocessing to deep learning. [S.l.]: O'Reilly Media, Inc, 2018.

ALENCAR, Ana C. D. **Inteligência Artificial, Ética de Direito**. São Paulo: Expressa, 2022.

ALLOGHANI, Mohamed *et al.* A systematic review on supervised and unsupervised machine learning algorithms for data science. **Supervised and unsupervised learning for data science**, 2020. p. 3-21.

ALTHUWAYNEE *et al.* Uncertainty reduction of unlabeled features in landslide inventory using machine learning t-SNE clustering and data mining apriori association rule algorithms. **Applied Sciences, v. 11, n. 2**, 2021. 556.

CÂMARA DOS DEPUTADOS. **Câmara dos Deputados**, 16 de Julho de 2022. Disponível em: https://www.camara.leg.br/noticias/809660-pandemia-acelera-o-uso-de-servicos-publicos-digitais.

CHATTERJEE, Soumick; JOSE, Pramod G.; DATTA, Debabrata. Text classification using SVM enhanced by multithreading and CUDA. **International Journal of Modern Education and Computer Science, v. 11, n. 1**, 2019. 11.

CHO, Danbi; LEE, Hyunyoung; KANG, Seungshik. An empirical study of Korean sentence representation with various tokenizations. **Electronics, v. 10**, 2021. 845.

CLAVIÉ, Benjamin; ALPHONSUS, Marc. The Unreasonable Effectiveness of the Baseline: Discussing SVMs in Legal Text Classification. **J Mundi - arXiv preprint arXiv:2109.07234**, 2021.

CNJ. **Manual de Utilização das Tabelas Processuais Unificadas do Poder Judiciário.** Brasília. 2014.

CNJ. Atos do CNJ. **Conselho Nacional de Justiça**, 17 de Julho de 2022. Disponível em: https://atos.cnj.jus.br/atos/detalhar/167.

CNJ. Atos do CNJ. **Conselho Nacional de Justiça**, 17 de Julho de 2022. Disponível em: https://atos.cnj.jus.br/atos/detalhar/3453.

CNJ. Balcão Virtual. **Conselho Nacional de Justiça**, 16 de Julho de 2022. Disponível em: https://www.cnj.jus.br/tecnologia-da-informacao-e-comunicacao/justica-4-0/balcao-virtual.

CNJ. Notícias do Conselho Nacional de Justiça. **Conselho Nacional de Justiça**, 16 de Julho de 2022. Disponível em: https://www.cnj.jus.br/tecnologia-da-informacao-e-comunicacao/justica-4-0/projeto-juizo-100-digital.

CNJ. Painel Justiça em Números. **Painel Justiça em Números**, 16 de Julho de 2022. Disponível em: https://paineis.cnj.jus.br/QvAJAXZfc/opendoc.htm?document=qvw_l%2FPainelCNJ.qvw&host=QVS%40neodimio03&anonymous=true&sheet=shResumoDespFT.

EASTERBROOK *et al.* Selecting Empirical Methods for Software Egineering Research. In: _____ **Guide to advanced empirical software engineering**. London: Springer, 2008. p. 285-311.

FACELI *et al.* **Inteligência artificial:** uma abordagem de aprendizado de máquina. 1ª edição. ed. [S.l.]: LTC, 2011.

FGV. **Tecnologia Aplicada à Gestão dos Conflitos no Âmbito do Poder Judiciário Brasileiro**. FGV Conhecimento - Centro de Inovação, Administração e Pesquisa do Judiciário. [S.l.]. 2020.

FGV. **RELATÓRIO DE PESQUISA: TECNOLOGIA APLICADA À GESTÃO DOS CONFLITOS NO ÂMBITO DO PODER JUDICIÁRIO - 2A FASE**. FGV - CIAPJ. [S.l.]. 2022.

GAGNO, L.P; BUFON, F.P. O processo coletivo e a suspensão dos processos individuais: uma análise conforme o direito fundamental de acesso à justiça. **a. Revista Eletrônica de Direito Processual**, 2020.

GERON, A. **Mãos à Obra:** Aprendizado de Máquina com Scikit-Learn & TensorFlow. Rio de Janeiro: Alta Books Editora, 2019.

GUO, Yan; WANG, Minxi; LI, Xin. Application of an improved Apriori algorithm in a mobile e-commerce recommendation system. **Industrial Management & Data Systems, v. 117, n. 2**, 2017. 287-303.

IZBICKI, R.l; DOS SANTOS, T. M. **Aprendizado de máquina:** uma abordagem estatística. 1a. ed. [S.l.]: [S.n.], 2020.

JAIN, Achin; JAIN, Vanita. Efficient Framework for Sentiment Classification Using Apriori Based Feature Reduction.**. EAI Endorsed Transactions on Scalable Information Systems, v. 8, n. 31**, 2021. e3.

KRISNANTO *et al.* Utilizing Apriori Data Mining Techniques on Sales Transactions. **Webology, v. 19, n. 1**, 2022. 5581-5590.

LI, Hongchan; YAO, Ni. Four-Layer Feature Selection Method for Scientific Literature based on Optimized K-Medoids and Apriori Algorithms. **International Journal of Performability Engineering, v. 15, n. 4**, 2019. 1141.

MA *et al.* Connections between Various Disorders: Combination Pattern Mining Using Apriori Algorithm Based on Diagnosis Information from Electronic Medical Records. **BioMed Research International, v. 2022**, 2022.

MAHMOOD, S.; SHAHBAZ, M.; GUERGACHI, A. Negative and positive association rules mining from text using frequent and infrequent item sets. **Scientific World Journal**, 2014.

MAKREHCHI, Masoud; KAMEL, Mohamed S. Extracting domain-specific stopwords for text classifiers. **Intelligent Data Analysis, v. 21**, 2017. 39-62.

MANJUNATH *et al.* Smart question answering system using vectorization approach and statistical scoring method. **Materials Today: Proceedings**, 2021.

NI, Jing; GAO, Ge; CHEN, Pengyu. Chinese text auto-categorization on petro-chemical industrial processes.. **Cybernetics and Information Technologies, v. 16, n. 6**, 2016. 69-82.

PANDAS. Package overview - Pandas. **Pandas**, 2022. Disponível em: https://pandas.pydata.org/docs/getting_started/overview.html. Acesso em: 28 Setembro 2022.

RÁCZ, Anita; BAJUSZ, Dávid; HÉBERGER, Károly. Effect of dataset size and train/test split ratios in QSAR/QSPR multiclass classification. **Molecules, v. 26**, 2021. 1111.

SAMUEL, A. L. Some studies in machine learning using the game of checkers. **Annual review in automatic programming**, 1969. v. 6, p. 1-36.

SARICA, Serhad; LUO, Jianxi. Stopwords in technical language processing. **Plos one, v. 16, n. 8**, 2021. e0254937.

SCIKIT-LEARN. Scikit-Learn - Cross Validation. **Scikit-Learn**, 2022. Disponível em: https://scikit-learn.org/stable/modules/cross_validation.html. Acesso em: 6 Setembro 2022.

SILVEIRA, R. *et al.* Topic modelling of legal documents via legal-bert. **Proceedings http://ceur-ws org ISSN**, 2021. v. 1613, p. 0073.

SOUSA, R. N. D. **MINERJUS: solução de apoio à classificação processual com uso de Inteligência Artificial.** TJTO. [S.l.]. 2019.

STJ. **Manual de padronização de textos do STJ**. Brasília. 2016.

STJ. Notícias antigas do STJ. **Superior Tribunal de Justiça**, 17 de Julho de 2022. Disponível em: https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias-antigas/2016/2016-03-02_20-07_Pleno-do-STJ-define-que-o-novo-CPC-entra-em-vigor-no-dia-18-de-marco.aspx.

TEGEGNIE, Alemu K.; TAREKEGN, Adane N.; ALEMU, Tamir A. A Comparative Study of Flat and Hierarchical Classification for Amharic News Text Using SVM. **International Journal of Information Engineering & Electronic Business, v. 9, n. 3**, 2017.

THEODORO JUNIOR, H. **Novo Código de Processo Civil Anotado 20ª Edição.** Rio de Janeiro-RJ: Editora forense, 2016.

TJPB. Notícias do TJPB. **Tribunal de Justiça da Paraíba**, 16 de Julho de 2022. Disponível em: https://www.tjpb.jus.br/noticia/atermacao-eletronica-permite-que-demandas-do-cidadao-sejam-resolvidas-sem-deslocamento-aos.

WANG *et al.* Predicting Self-Reported Proactive Personality Classification With Weibo Text and Short Answer Text. **IEEE Access, v. 9**, 2021. 77203-77211.

Apêndices

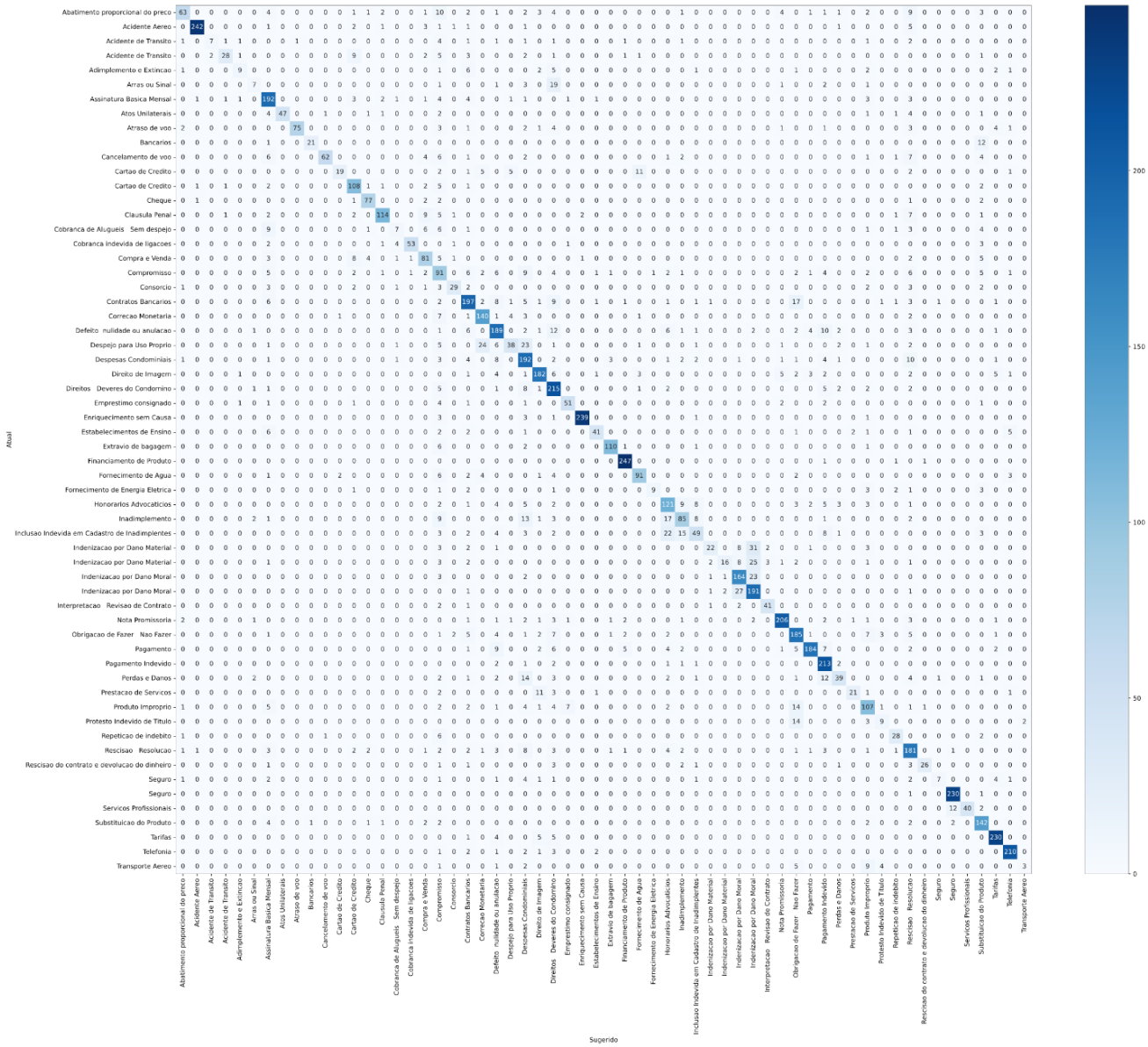
APÊNDICE A – Extensão das *stopwords*

Lista não exaustiva de *stopwords* geradas com base nos documentos jurídicos analisados durante o processamento do *corpus* utilizado nesta pesquisa para extensão das *stopwords* pré-definidas pela biblioteca NLTK do python na língua portuguesa.

'agravo', 'ainda', 'além', 'ante', 'apelação', 'art', 'artigo', 'assim', 'autor', 'autora', 'ação', 'caput', 'caso', 'causa', 'cinco', 'civil', 'cláusulas', 'conforme', 'custas', 'cível', 'código', 'deferido', 'desde', 'desta', 'destas', 'deste', 'destes', 'deve', 'dever', 'dez', 'diante', 'direito', 'dois', 'entendimento', 'então', 'excelência', 'expor', 'exposto', 'fato', 'fatos', 'fim', 'indeferido', 'inicial', 'judicial', 'judiciária', 'juiz', 'julgamento', 'junto', 'juridica', 'justiça', 'juízo', 'legal', 'lei', 'mil', 'nada', 'naquele', 'naqueles', 'neste', 'nestes', 'nestes', 'nove', 'oito', 'ora', 'outra', 'parte', 'parágrafo', 'passa', 'pedido', 'pode', 'pois', 'princípio', 'processo', 'processuais', 'processual', 'promovente', 'promovida', 'promovido', 'prova', 'qualquer', 'quatro', 'razão', 'reais', 'relator', 'requer', 'segue', 'seguinte', 'segundo', 'seis', 'sendo', 'ser', 'sete', 'sob', 'sobre', 'tal', 'ter', 'termos', 'tj', 'toda', 'todas', 'todo', 'todos', 'tribunal', 'três', 'tudo', 'um', 'valor', 'vejamos', 'vez', 'vossa'

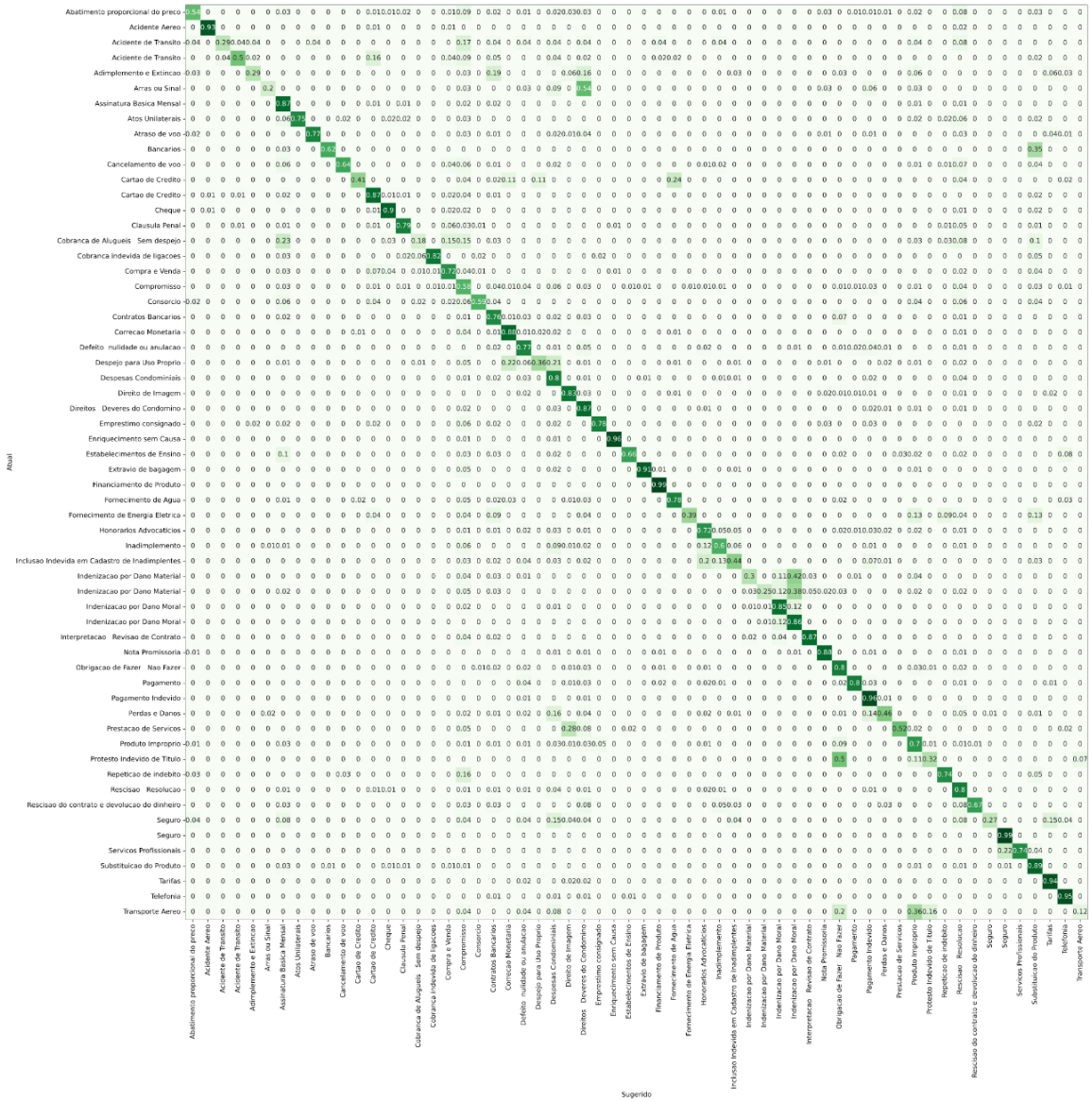
APÊNDICE B – Matrizes de confusão *Apriori/SVM*

Matriz de confusão *Apriori/SVM*



Fonte: Desenvolvido pelo autor.

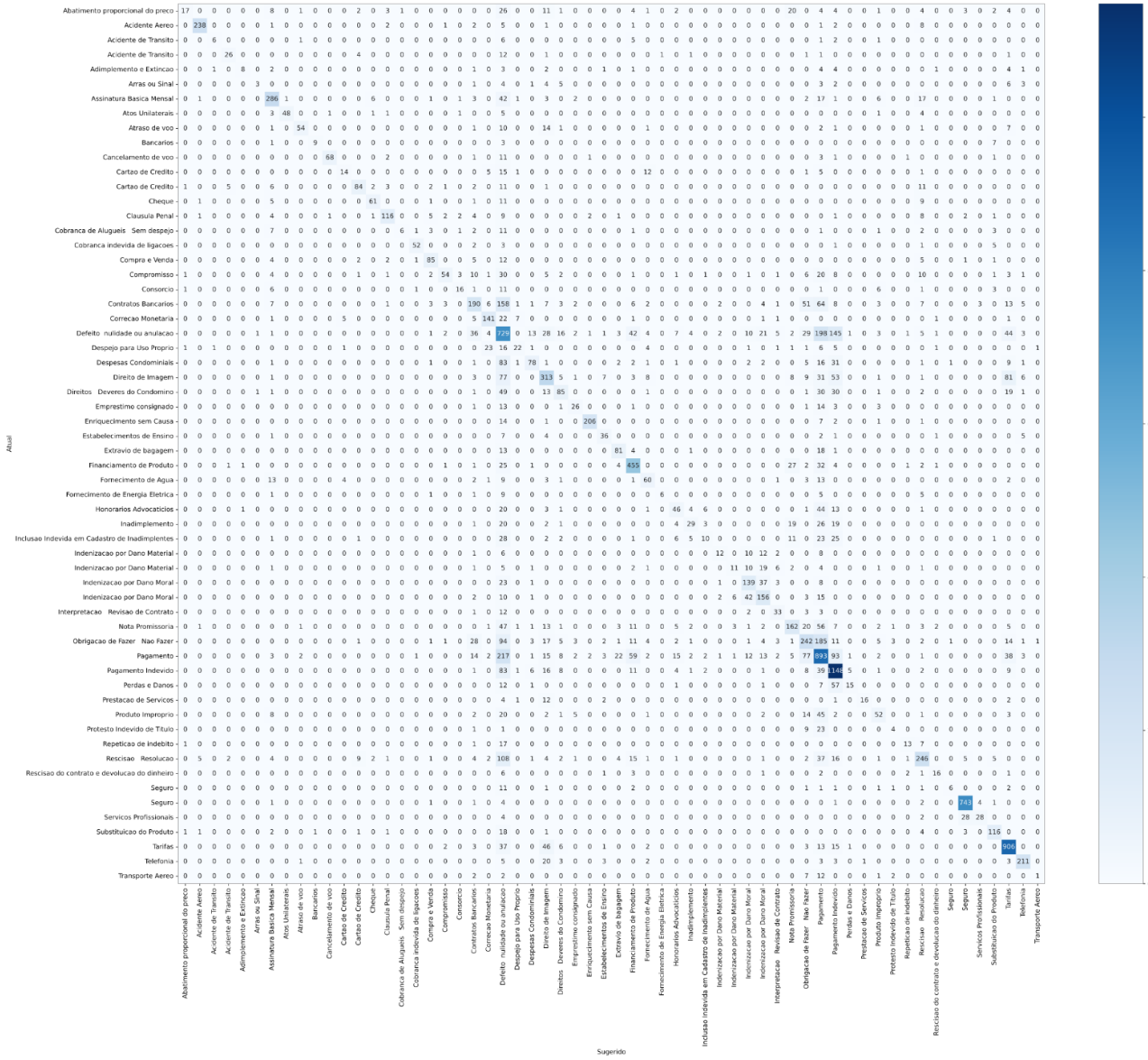
Matriz de confusão normalizada *Apriori/SVM*



Fonte: Desenvolvido pelo autor.

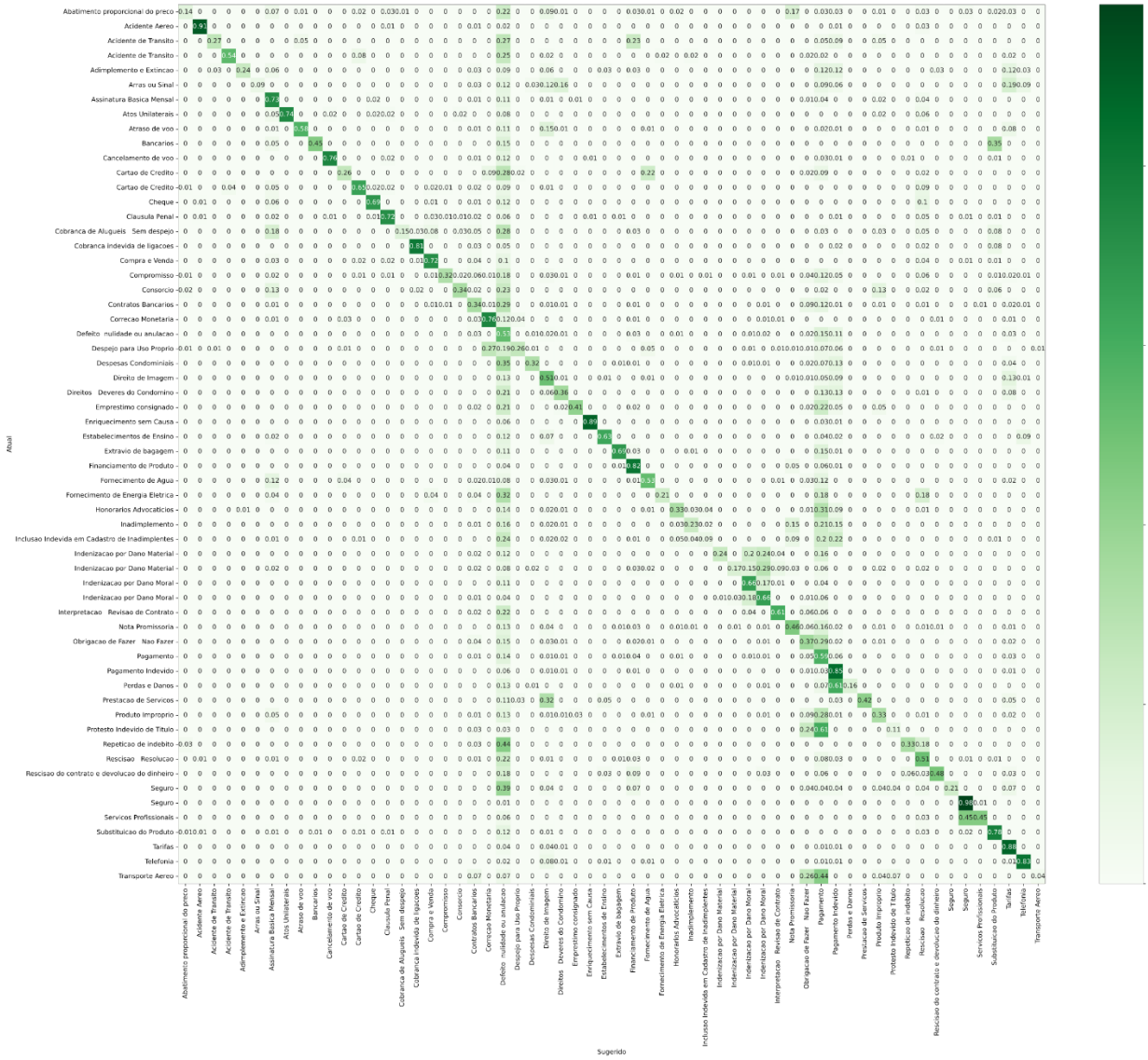
APÊNDICE C – Matrizes de confusão *SVM*

Matriz de confusão *SVM*



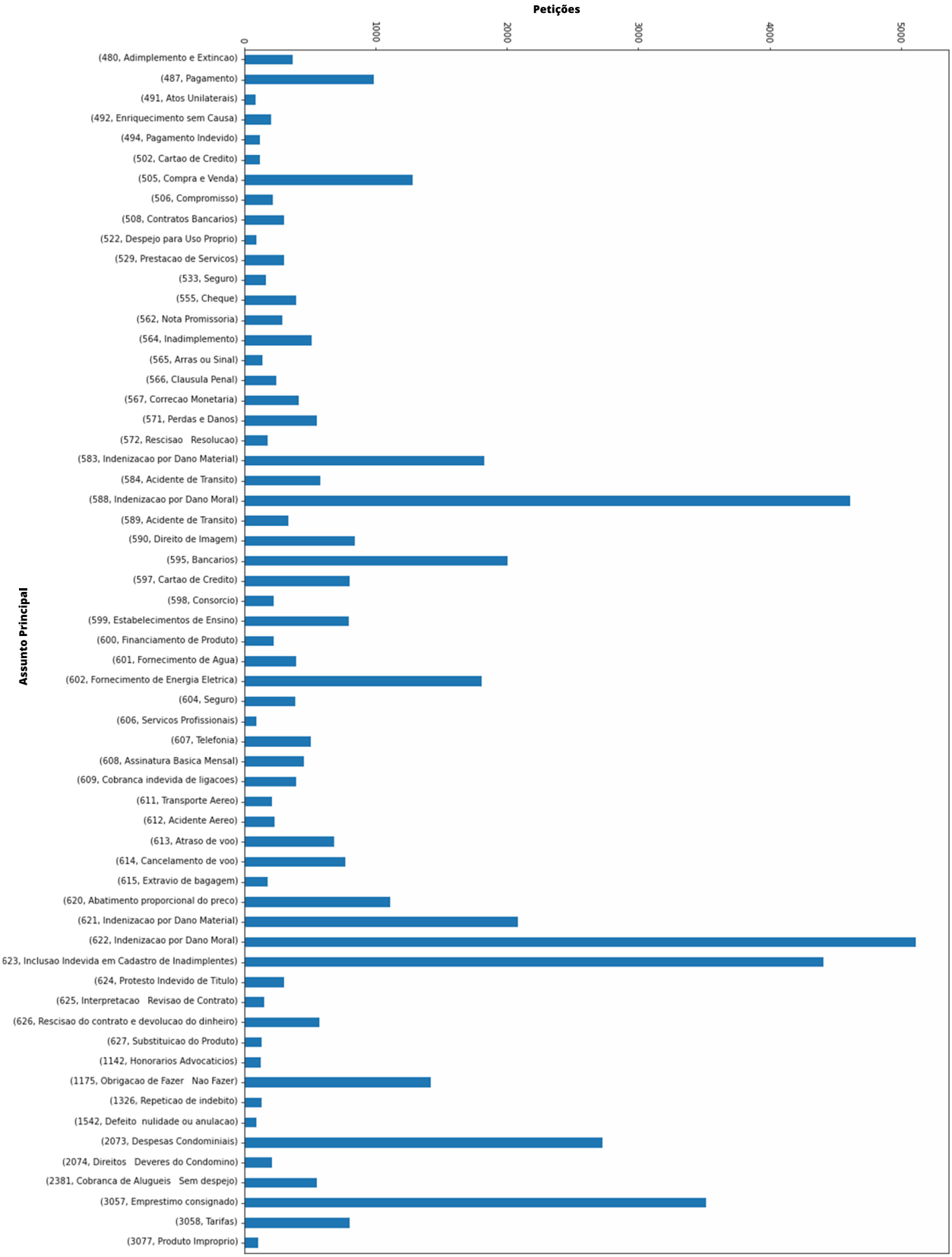
Fonte: Desenvolvido pelo autor.

Matriz de confusão normalizada *SVM*

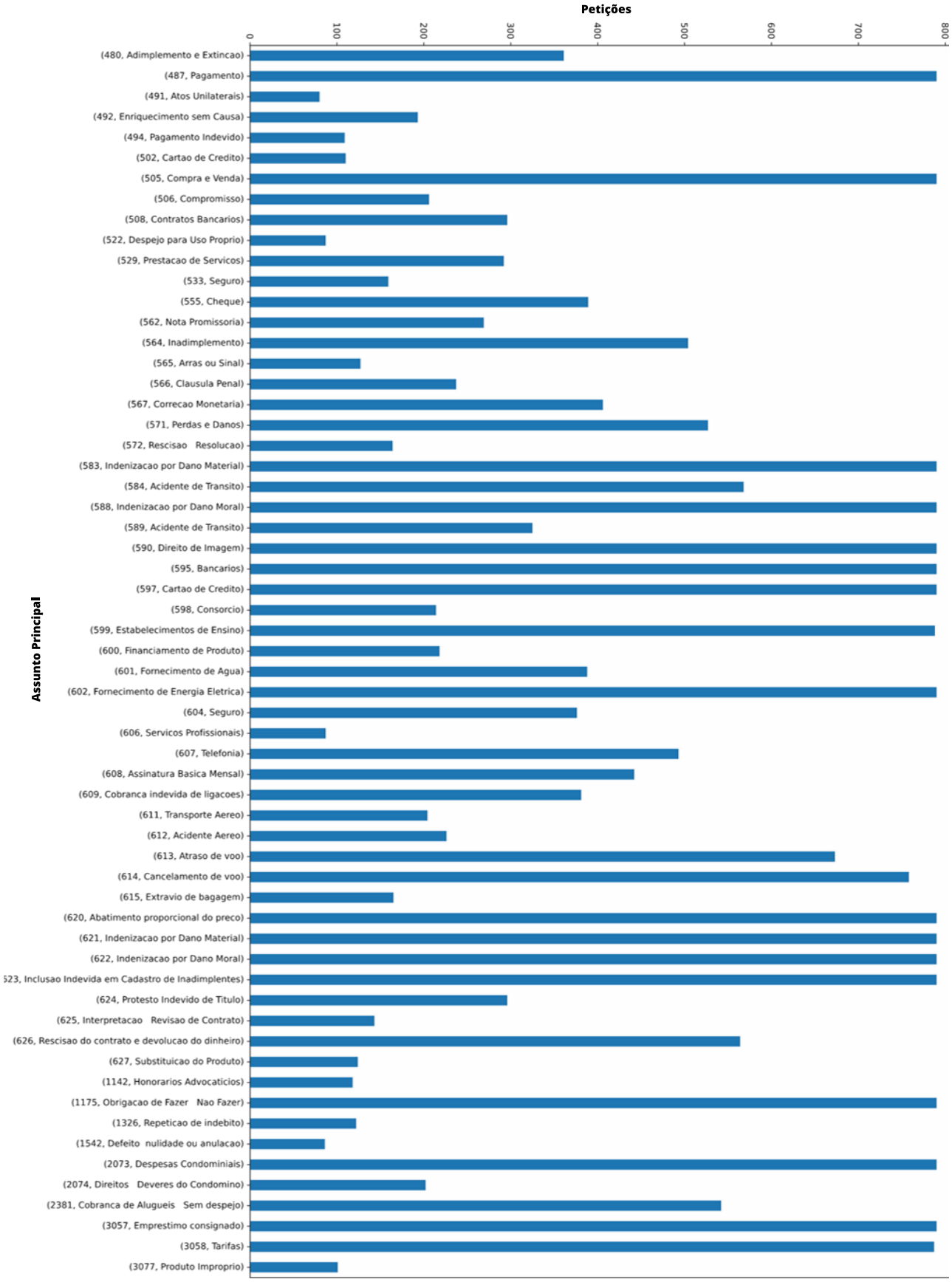


Fonte: Desenvolvido pelo autor.

APÊNDICE D – Petições por assunto pré refinamento de associação



APÊNDICE E – Petições por assunto pós refinamento de associação



ENTREGA DA VERSÃO FINAL DE DISSERTAÇÃO

Eu, PROF. Dra. Juliana Dantas Ribeiro Viana de Medeiros, autorizo o aluno(a) Samuel de Aguiar Rodrigues a entregar a versão final da dissertação de mestrado, à secretaria do PPGTI, que foi por mim analisada e está de acordo com os apontamentos feitos pelos membros da banca de apresentação do referido aluno.

`

Prof. Dra. Juliana Dantas Ribeiro Viana de Medeiros
Orientador

João Pessoa, 14 de Outubro de 2022.