UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM DE ENGENHARIA DA COMPUTAÇÃO E SISTEMAS

ANÁLISE E AVALIAÇÃO DE TAREFAS PARA PRÉ-PROCESSAMENTO EM MINERAÇÃO DE TEXTOS JURÍDICOS

MARCOS VINICIUS JANUÁRIO DA SILVA

SÃO LUÍS - MA
2023

UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO E SISTEMAS

MARCOS VINICIUS JANUÁRIO DA SILVA

ANÁLISE E AVALIAÇÃO DE TAREFAS PARA PRÉ-PROCESSAMENTO EM MINERAÇÃO DE TEXTOS JURÍDICOS

Dissertação apresentada ao curso de Mestrado Profissional em Engenharia da Computação e Sistemas na Universidade Estadual do Maranhão como pré-requisito para obtenção do título de Mestre sob orientação do Prof. Msc. Antonio Fernando Lavareda Jacob Jr e co-orientação do Prof. Dr. Fábio Manoel França Lobato.

SÃO LUÍS - MA
2023

Silva, Marcos Vinicius Januário da
Análise e avaliação de tarefas para pré-processamento em mineração de textos jurídicos / Marcos Vinicius Januário da Silva. – São Luis, MA, 2023.
69 f

Dissertação (Mestrado em Engenharia de Computação e Sistemas) – Universidade Estadual do Maranhão, 2023.

Orientador: Prof. Me. Antonio Fernando Lavareda Jacob Jr

1.Mineração de Textos. 2.Pré-Processamento. 3.Dados Jurídico. I.Título

CDU:004.451.5

Elaborado por Cássia Diniz- CRB 13/910

MARCOS VINICIUS JANUÁRIO DA SILVA

ANÁLISE E AVALIAÇÃO DE TAREFAS PARA PRÉ-PROCESSAMENTO EM MINERAÇÃO DE TEXTOS JURÍDICOS

Dissertação apresentada ao curso de Mestrado Profissional em Engenharia da Computação e Sistemas na Universidade Estadual do Maranhão como pré-requisito para obtenção do título de Mestre sob orientação do Prof. Msc. Antonio Fernando Lavareda Jacob Jr e co-orientação do Prof. Dr. Fábio Manoel França Lobato.

Aprovada em: 15/09/2023.

BANCA EXAMINADORA

Documento assinado digitalmente
gov.br ANTONIO FERNANDO LAVAREDA JACOB JUNIOI
Data: 30/11/2023 08:54:46-0300
Verifique em https://validar.iti.gov.br

Prof. Me. Antônio Fernando Lavareda Jacob Junior
Orientador - PECS UEMA

Documento assinado digitalmente
gov.br FABIO MANOEL FRANCA LOBATO
Data: 29/11/2023 11:12:46-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Fábio Manoel França Lobato
Co-orientador - PECS UEMA/UFOPA

Documento assinado digitalmente
gov.br EWALDO EDER CARVALHO SANTANA
Data: 29/11/2023 10:23:45-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Ewaldo Eder Carvalho Santana
Membro Interno - PECS UEMA

RANIEL BARBOSA NUNES:00999178318
Assinado de forma digital por RANIEL BARBOSA NUNES:00999178318
Dados: 2023.11.28 11:14:36 -03'00'

MM. Juiz Raniel Barbosa Nunes
Membro Externo - TJMA

Documento assinado digitalmente
gov.br FRANCISCO DE ARAUJO COSTA
Data: 27/11/2023 12:58:32-0300
Verifique em https://validar.iti.gov.br

Francisco De Araújo Costa
Membro Externo - TJMA

SÃO LUÍS - MA

2023

AGRADECIMENTOS

Em primeiro lugar agradeço a Deus, por sempre ter me guiado, me protegido e me dado sabedoria. Sei que sem Ele eu não estaria realizando este trabalho e nem estaria vivo.

Tenho também a honra de estar ao lado de pessoas que me apoiam e me ajudam. Minha família sempre me incentivou a prosseguir nos estudos e assim consegui chegar a esta etapa da minha vida.

Sem falar em amizades que nos fortalecem e nos dão ânimo para caminhar. Em vários momentos da vida são as amizades que nos fazem permanecer e, por isso, agradeço por cada amigo que tem estado comigo em momentos bons e difíceis.

Agradeço também ao meu orientador, Prof. Me. Antônio Fernando Lavareda Jacob Junior, por ter permitido e aceito que trabalhássemos juntos neste projeto. Agradeço todo ensino e suporte dado por ele.

A todos os que acompanharam este processo e de alguma forma me inspiraram, auxiliaram e incentivaram para que eu pudesse concluir este projeto, meus sinceros agradecimentos.

Finalmente, agradeço a cada leitor que dedicou um pouco de seu tempo na leitura deste trabalho. Espero que seja de grande valia para suas vidas.

Assim, agradeço a todos vocês. Obrigado!

Dedico este trabalho para a minha família,
que em todos os momentos esteve me
apoiando e me ajudando a não desistir.

RESUMO

O Aprendizado de Máquina tem sido empregado em diversas áreas do conhecimento, agregando grandes benefícios, seja facilitando tarefas complexas para minimizar gastos e tempo em soluções ou encontrando informações relevantes não evidenciadas por olhares humanos em conjuntos de dados. A Descoberta de Conhecimento em Base de Dados representa uma ferramenta bastante empregada, onde a Mineração de Textos é utilizada para gerar conhecimento a partir de dados contendo elementos textuais, como é o caso no contexto jurídico. O âmbito judiciário possui um vocabulário próprio amplamente rico, com termos específicos que não são utilizados em outros contextos do português brasileiro. É necessário que estes elementos textuais estejam devidamente tratados antes de serem empregados no processo de mineração. O Pré-Processamento é uma tarefa essencial para aumentar a relevância dos dados estudados, removendo ou modificando estruturas no texto. Porém, existem diversas técnicas na preparação dos dados que podem ser executadas, cada uma fornecendo ganhos diferentes para o processo. Dependendo do cenário, uma tarefa pode ser fundamental para aumentar a relevância, enquanto outra pode não produzir nenhum efeito. Nesse contexto, o presente trabalho propõe realizar uma análise das principais tarefas de Pré-Processamento empregadas em Mineração de Textos Jurídicos. Para atingir esse objetivo, foram realizados uma série de experimentos com a combinação de técnicas de pré-processamento, tendo como estudo de caso um classificador de precedentes e como métrica de avaliação a acurácia. Por mim, conclui-se que as técnicas Stemização, Lematização e Tokenização obtiveram os melhores resultados no contexto.

Palavras-chave: *Mineração de Textos. Pré-Processamento. Dados Jurídico.*

ABSTRACT

Machine Learning has been utilized in various knowledge domains, offering significant benefits. It facilitates complex tasks, minimizing costs and time in solutions, and identifies relevant information not evident to human eyes in data sets. The discovery of knowledge in databases represents a widely used tool. The judiciary possesses a rich vocabulary unique to its context, featuring specific terms not used in other Brazilian Portuguese contexts. Proper treatment of these textual elements is necessary before utilizing them in the mining process. Preprocessing is essential to enhance the relevance of the studied data by removing or modifying structures in the text. Nevertheless, various data preparation techniques can be employed, each contributing different gains to the process. Depending on the scenario, a task can be crucial for increasing relevance, while another may not produce any effect. In this context, the present work proposes to analyze the main Preprocessing tasks employed in Legal Text Mining. To achieve this goal, a series of experiments combined preprocessing techniques with a precedent classifier as a case study and accuracy as the evaluation metric. It was concluded that stemming, lemmatization, and tokenization techniques yielded the best results in this context

Keywords: *Text Mining. Preprocessing. Legal Data.*

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

BERT	*Bidirecional Encoder Representations from Transformers*
CNJ	Conselho Nacional de Justiça
CRISP-DM	*CRoss Industry Standard Process for Data Mining*
IA	Inteligência Artificial
KDD	*Knowledge Discovery in Databases*
LDA	*Latent Dirichlet Allocation*
LR	*Logistic Regression*
KNN	*K-Nearest Neighbor*
PLN	Processamento de Linguagem Natural
STF	Supremo Tribunal Federal
SVM	*Support Vector Machine*
TF-IDF	*Term Frequency–Inverse Document Frequency*

SUMÁRIO

1. INTRODUÇÃO

O Poder Judiciário Brasileiro finalizou o ano de 2021 com 77,3 milhões de processos em tramitação, aguardando alguma solução definitiva (CNJ, 2022). Dada essa quantidade crescente de processos, é perceptível como o Sistema Judiciário encontra-se sobrecarregado para atender todas as demandas pendentes.

Técnicas de Aprendizado de Máquinas, como classificação, podem auxiliar na categorização de temas em conjuntos de documentos, como petições iniciais (HAGEN, 2018). Com esta análise, entidades do setor jurídico podem acelerar a tomada de decisão, obtendo documentos relacionados definidos pelos temas categorizados na classificação.

A etapa de pré-processamento é crucial na mineração de textos para garantir a qualidade dos dados utilizados em tarefas como classificação (WIRTH; HIPP, 2000). Ela envolve a limpeza, estruturação e organização dos documentos, proporcionando ganhos significativos nos resultados finais (GRANCHAROVA; JANGEFALK, 2018). Mesmo com as melhores técnicas e parâmetros, o tratamento adequado dos dados é fundamental para evitar erros e maximizar a eficácia da análise (Ribeiro et al. 2020). A constante pesquisa na área e a construção de Bases de Dados contribuem para o aperfeiçoamento contínuo dessa etapa (SILVA et al. 2021).

Nesse contexto, esse trabalho analisa a utilização de técnicas de pré- processamento de textos voltados para a mineração de dados jurídicos, aplicando as técnicas em documentos de petições iniciais e empregando modelos de Classificação, propondo um fluxo de pré-processamento construído a partir dos estudos realizados. A pesquisa permite identificar as melhores combinações de técnicas a serem aplicadas para os documentos mencionados, contribuindo assim para que futuros estudos na área de mineração de textos possam utilizar desse fluxo de pré-processamento.

1.1. JUSTIFICATIVA

Diversas são as tarefas que podem fazer parte do processo de Preparação de Dados e cada uma fornece ganhos significativos para a mineração de forma geral se aplicadas corretamente. Uma mesma tarefa, executada em processos distintos com dados diferentes, apresenta efeitos diferentes para cada caso, podendo ser de fundamental importância para aumentar a relevância dos dados em um processo e não produzir nenhum ganho relevante em outro.

É notório que não há um padrão estruturado de definição das tarefas a serem escolhidas e de ordem de execução destas na etapa de Pré-Processamento no processo de Descoberta de Conhecimento e em Mineração de Dados (CIRQUEIRA et al., 2017). Assim, cada pesquisa e projeto desenvolve seu processo de preparação de dados de maneira própria, fazendo uso de várias tarefas para tentar alcançar um fluxo adequado de processamento dos dados.

Tanto os profissionais quanto os pesquisadores do ramo estão em busca de garantir resultados confiáveis e precisos por técnicas de pré-processamento de dados equilibrando eficiência e complexidade de tempo, já que a preparação de dados pode levar uma quantidade considerável de tempo de processamento (GARCÍA et al., 2015).

Em problemas de aprendizado de máquina, as tarefas de limpeza, preparação e filtragem de dados podem levar até 80% do tempo no pré-processamento de dados em projetos de mineração de dados em um contexto real. Com isso, o pré-processamento de conjuntos de dados executado da maneira mais eficiente permanece sendo um fator fundamental para problemas de mineração de dados (SALEEM et al., 2014).

Dessa forma, o presente trabalho propõe realizar uma análise e avaliação das principais tarefas utilizadas em Mineração de Textos aplicada para Dados Jurídicos, seguindo um fluxo de execução de testes para as tarefas associadas.

1.2. OBJETIVOS

OBJETIVO GERAL

Analisar as principais técnicas de Pré-Processamento de Mineração de Textos voltadas para a Área Jurídica e com isso realizar a avaliação e implementação das melhores técnicas para elaborar um fluxo de pré-processamento construído a partir dos estudos realizados.

OBJETIVOS ESPECÍFICOS

- Realizar uma revisão bibliográficas das técnicas de pré-processamento de textos aplicadas à análise de textos jurídicos;
- Implementar as principais técnicas de pré-processamento de textos utilizadas no domínio jurídico; Realizar experimentos utilizando as principais técnicas encontradas;

- Avaliar o impacto dos métodos de pré-processamento de textos em análises relevantes ao domínio de aplicação;
- Propor um fluxo de pré-processamento otimizado a partir dos experimentos realizados.

1.3. ESTRUTURA DO DOCUMENTO

Este trabalho está estruturado da seguinte forma: no capítulo 2 é apresentado o Referencial Teórico; o Capítulo 3 expõe o Framework Experimental; no Capítulo 4 são discutidos os resultados obtidos; por fim, no Capítulo 5 é apresentada a Conclusão.

2. REFERENCIAL TEÓRICO

Esta seção apresenta os principais conceitos e embasamentos teóricos para a construção deste Projeto. Inicialmente, são discutidos temas significativos, como Aprendizado de Máquina e Mineração de Dados. Prosseguindo, são destacados os conteúdos relativos à estrutura de Textos Jurídicos e às Tarefas de Preparação de Dados, com as principais Etapas em Pré-Processamento. Também são apresentados Trabalhos Relacionados à Pesquisa em Desenvolvimento.

2.1. Aprendizado de Máquina e Mineração de Dados

O objetivo do Aprendizado de Máquina consiste em identificar padrões presentes em grandes volumes de dados (HAGEN, 2018). Estes dados podem representar diversos elementos, podendo ser numéricos, textuais ou conjuntos destes. Com os padrões identificados é possível que o sistema tome decisões, automatizando tarefas com o mínimo de intervenção de seres humanos e constrói sua própria lógica com os dados fornecidos (FREITAS JUNIOR, 2020).

Para isso, cada sistema pode ser formatado para trabalhar de maneiras diferentes. De maneira geral, pode-se dividir os sistemas de Aprendizado de Máquinas em três subcategorias, dependendo de como ele utilizará as informações e quais serão as possíveis intervenções de análise humana, se existirem. Estas três categorias são: Aprendizado Supervisionado, Aprendizado Não-Supervisionado e Semi-Supervisionado.

O Aprendizado Supervisionado possui regras bem definidas baseadas em rótulos e classes estipuladas anteriormente para que o sistema conheça os valores verdadeiros para o conjunto de dados de observações estudado (ALMEIDA, 2019). Dessa forma, os resultados encontrados seguem um padrão conhecido, sendo possível estipular o comportamento que será apresentado (GONÇALVES, 2020). Alguns exemplos de Aprendizado Supervisionado são a Classificação e a Regressão.

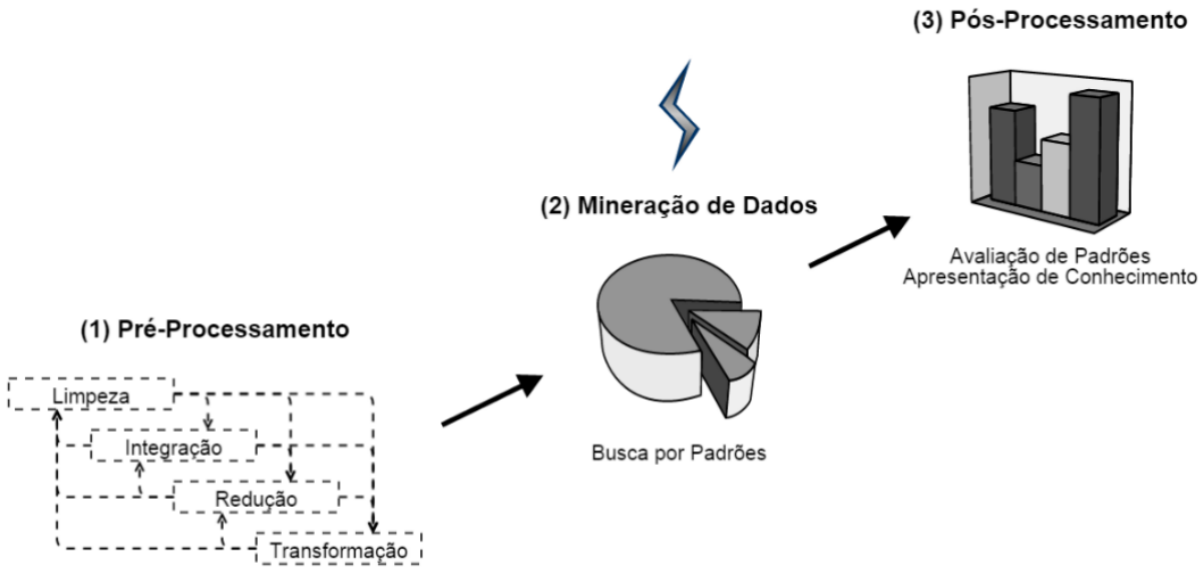
Diferentemente da categoria descrita, no Aprendizado Não-Supervisionado não são conhecidos os apontamentos que o sistema irá representar, mesmo que os dados fornecidos possuam estrutura conhecida. Não é possível determinar os valores que o sistema irá apresentar como resposta para o conteúdo dado como entrada (GONÇALVES, 2020). O sistema busca por padrões ocultos para organizar os dados de forma a fazer sentido ou se

tornar útil para a aplicação alvo. A técnica de Agrupamento é um exemplo prático deste tipo de Aprendizado (ALMEIDA, 2019).

Por conseguinte, sistemas Semi-Supervisionados possuem por principal característica mesclar as duas metodologias apresentadas anteriormente. Sendo uma abordagem híbrida utiliza-se de classes definidas para se obter informações sobre o problema e seguir o processo de forma não-supervisionada com os dados não rotulados (ALMEIDA, 2019).

Como apontado, o Aprendizado de Máquina consiste em descrever padrões em conjuntos de dados. Para isso, a Descoberta de Conhecimento em Base Dados (*Knowledge Discovery in Database*, KDD), propõe técnicas para utilizar da maneira mais eficiente todos os recursos disponíveis para as tarefas de transformação de dados brutos em conhecimento útil, assim compreendendo os dados (RAMINELLI; SANTOS, 2019). É possível inserir a Mineração de Dados dentro do processo de KDD, seguindo assim as etapas: Pré-Processamento (primeiro tratamento nos dados, realizando limpeza, integração, redução e transformação aplicando técnicas apropriadas), Mineração de Dados (utilização de algoritmos específicos para realizar a busca por padrões) e Pós-Processamento (análise e interpretação para apresentação compreensível ao ser humano, avaliando os padrões gerados pela Mineração de Dados) (MACIEL et al., 2015).

Figura 1. Mineração de Dados no Processo de Descoberta de Conhecimento.



Fonte: Maciel et al. (2015).

A Mineração de Dados, como descrito, torna-se uma etapa fundamental no KDD. Ela se dá pela utilização de técnicas e algoritmos de Aprendizado de Máquina obtendo informações relevantes em Banco de Dados projetados anteriormente (RAMINELLI; SANTOS, 2019).

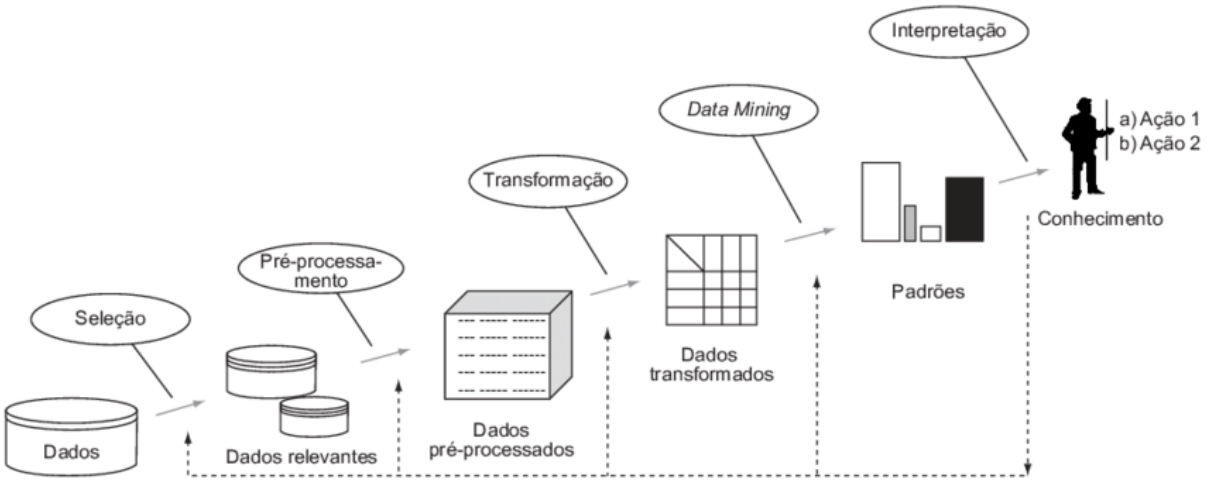
2.2. Mineração de Textos Jurídicos

É necessário que os sistemas sejam capazes de se adequar para qualquer aplicação que seja alvo de estudo. A Mineração de Dados abrange casos voltados para diversas áreas, como Medicina, Sistemas de Recomendação e Comércio. Da mesma maneira, pesquisas estão sendo implementadas para a aplicação de Aprendizado de Máquina no Contexto Judiciário.

Os textos jurídicos possuem regras estéticas e elementos próprios ao contexto judiciário. Assim, compreender as melhores técnicas a serem utilizadas é fundamental para que os dados fornecidos possam ser tratados e utilizados de maneira eficiente pelos algoritmos desenvolvidos (MACHADO, 2019).

A Mineração de Textos consiste em um processo de Descoberta de Conhecimento que utiliza como base os elementos textuais, dados não estruturados ou semiestruturados, para a extração e análise (MADEIRA, 2015). A Mineração de Textos inicia com a seleção dos dados, definindo tamanhos e espaços que serão utilizados, passando pelo Pré-Processamento e Transformação para que então os algoritmos de mineração encontrem os padrões desejados e seja possível posteriormente uma interpretação clara dos resultados (FAYYAD, 1996).

Figura 2. Processo de Descoberta de Conhecimento e Mineração de Dados.



Fonte: Fayyad (1996).

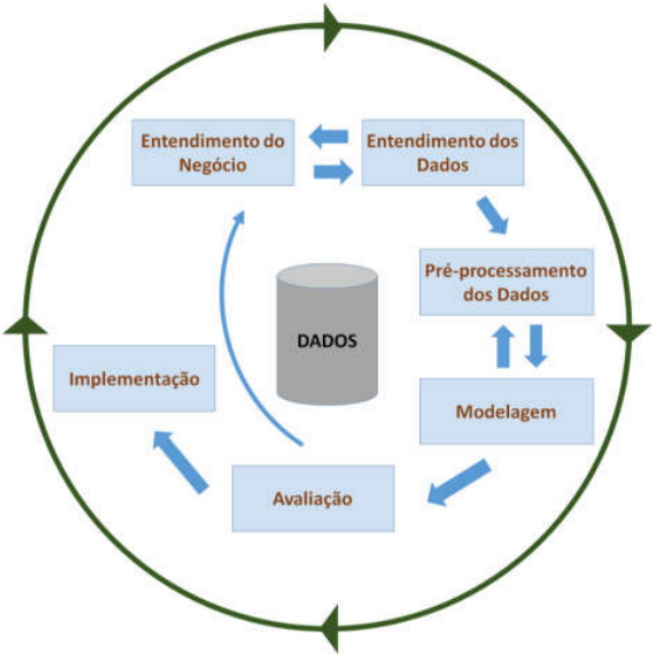
Dessa forma, para que a Mineração de Textos consiga desempenhar de maneira eficaz seu processo é preciso que os passos anteriores sejam bem definidos e estruturados. A etapa de Pré-Processamento pode agilizar o processamento na mineração de processos, impactando diretamente na qualidade dos modelos desenvolvidos para este propósito (D'CASTRO, 2020).

Dessa forma, se torna essencial que os sistemas sejam capazes de se adaptar para o âmbito jurídico, tratando os dados textuais provenientes desses processos com as tarefas de Pré-Processamento adequado, como concorda (D'CASTRO, 2020).

2.3. Tarefas de Pré-Processamento

O CRISP-DM (*CRoss Industry Standard Process for Data Mining*, ou Processo Padrão Inter-Industrial para Mineração de Dados) como uma metodologia eficaz para descrever e projetar as etapas que compõem o fluxo de desenvolvimento de sistemas e aplicações voltadas para a Mineração de Dados. As fases principais do CRISP-DM são: Entendimento do Negócio, Entendimento dos Dados, Preparação dos Dados, Modelagem, Avaliação e Implantação (WIRTH; HIPP, 2000).

Figura 3. Etapas da Metodologia CRISP-DM.



Fonte: Coelho et al. (2016).

A fase inicial do CRISP-DM, Entendimento do Negócio, é focada em observar os principais objetivos e requerimentos do projeto e realizar o planejamento inicial para que o projeto atinja as metas estabelecidas. Nesta fase é necessário converter o problema investigado em um problema de Data Mining, interagindo com especialistas da área em questão para que o entendimento seja coerente, também realizando um levantamento de possíveis bases de dados existentes de estudos anteriores que podem auxiliar o projeto. Esta fase requer, além disso, que seja realizada capacitação dos integrantes das equipes, para que haja nivelamento do conhecimento aplicado ao negócio proposto.

A fase de Entendimento dos Dados compreende tarefas que incluem busca e coleta de dados pelo levantamento realizado. Além de identificação de possíveis problemas e apresentação de hipóteses iniciais a partir dos dados que foram encontrados ou gerados. É necessário conhecer e entender os dados disponíveis para o processo antes de aplicar ao processo de Modelagem, avaliando a qualidade e o volume dos dados, podendo acontecer formação das primeiras hipóteses de informação relevante para o sistema.

A Preparação dos Dados dentro do CRISP-DM antecede a construção de modelos e inclui tarefas de seleção dos dados para a análise, limpeza e transformação dos dados, podendo ser efetuada várias vezes durante o processo. Nesta fase também são definidos novos atributos a partir dos dados, ocorrendo derivação de atributos existentes nas Bases de Dados. É possível também que a preparação compreenda técnicas de vetorização dos dados se os estes forem de origem textual, como frases, palavras ou documentos inteiros, transformando-os de texto para dados numéricos.

A Modelagem tem como objetivo realizar a seleção e aplicação das técnicas mais adequadas para o contexto, além de realizar ajustes nos parâmetros empregados dentro dos modelos criados. É preciso escolher e utilizar as técnicas adequadas para o problema, pois existem várias soluções para a mesma questão, onde uma pode ser fundamental para a solução que se busca, enquanto outra não gera ganhos relevantes.

A fase de Avaliação é constante e aponta para uma revisão dos passos que foram e estão sendo executados dentro do processo, realizando também a análise dos resultados que foram encontrados a partir da Modelagem. Com a avaliação é possível identificar se o modelo construído condiz com as expectativas estabelecidas, entendendo se os resultados podem estão aceitáveis ou apresentam necessidade de revisão, sendo importante que sempre existam interações com os especialistas da área.

A última fase do CRISP-DM, Implantação, compreende a organização e apresentação do conhecimento descoberto para que seja disponível para utilização pelo usuário final do sistema.

Todo o fluxo do CRISP-DM é executado de maneira cíclica, onde é possível retornar para uma fase anterior, se necessário, ou mesmo partir do princípio do processo até chegar às fases de Avaliação e Implantação. E, mesmo após a implantação, pode ser que este processo seja revisado para fornecer atualizações e novos conhecimentos a partir da inclusão de novos dados e técnicas.

Dentro deste Processo, é importante ressaltar que a etapa de Preparação dos Dados se torna de fundamental importância, incluindo tarefas voltadas para o Pré-Processamento e a Normalização dos dados. Esta etapa fornece ferramentas de limpeza e tratamento dos dados que serão utilizados pela mineração. O objetivo do Pré-Processamento consiste em obter as características chaves presentes nos dados e aumentar a relevância entre os termos e o aprendizado que será estabelecido (KADHIM, 2018).

A aplicação de Pré-Processamento adequado fornece ganhos significativos para todo o processo de Descoberta de Conhecimento. Partindo do princípio de que os dados serão utilizados posteriormente em algoritmos propriamente estabelecidos, é necessário que estes estejam preparados de maneira apropriada na etapa de Pré-Processamento, alavancando os resultados e reduzindo as falhas (RIBEIRO, 2020).

É possível citar algumas das tarefas mais utilizadas para a aplicação da etapa de pré-processamento, que estão presentes em diversos projetos em mineração de textos. Cada uma dessas técnicas é importante para uma finalidade específica, ficando à escolha dos desenvolvedores a sua possível utilização.

Uma técnica de processamento de textos bastante utilizada em grande parte das aplicações é a remoção de *stopwords*. O termo "*stopword*s" pode ser traduzido como "palavras de parada" em tradução livre. A função desta técnica se baseia em retirar do texto elementos que possuem frequência ou semântica tem pouco ou nenhum significado para a recuperação de informações.

Estes termos se tornam irrelevantes para o objetivo final da mineração empregada. Alguns exemplos de *stopwords* comuns que são removidos são: preposições, conjunções, artigos, advérbios, entre outros tipos de palavras que não irão gerar nenhum benefício para o resultado das métricas estudadas, ou muitas vezes prejudicam a modelagem construída.

Outra técnica utilizada para o pré-processamento é a remoção de caracteres especiais, além de pontuações e acentuações dentro do texto. Estes caracteres interferem nas métricas da

mineração realizada já que não fornecem nenhum ganho semântico para a interpretação pelo modelo e diminuindo a concentração de termos relevantes para a aplicação.

A stemização ("*stemming*" em inglês) tem o objetivo de reduzir a palavra-alvo para o seu radical. Como exemplo, podemos citar as palavras "garoto" e "garota" que seriam reduzidas a "garot". No caso da lematização ("*lemmatization*" em inglês) tem a função de reduzir a palavra ao seu lema, sendo a forma no masculino e singular da palavra. Para o caso de verbos, o lema é a forma infinitiva.

Por exemplo, para as palavras "garoto", "garota", "garotos" e "garotas" são todas reduzidas para o mesmo lema: "garoto". Como exemplo para verbos, as palavras "fizer", "faço" e "fiz" são todas transformadas para o lema "fazer". Tanto para a stemização quanto para a lematização a principal vantagem está em reduzir o vocabulário e abstração dos textos.

As tarefas de remoção de números e espaços em branco geram benefícios pois estes elementos não interferem na semântica estudada. Em alguns projetos não é interessante retirar os dígitos ou números, mas em casos em que não há influência destes elementos no contexto geral da mineração se torna eficiente a sua remoção.

A tokenização é uma tarefa utilizada como decomposição de um texto em cada termo que o compõe. Para verificar a separação entre cada termo ("*token*"), geralmente se utilizam alguns delimitadores para realizar a tokenização, como: espaços em branco e quebras de linhas. Porém, a tokenização pode ocorrer também com a divisão com dois termos (2-gram, bigrama) ou com três termos (3-gram, trigrama). Dessa forma, chama "n-grama" uma sequência de n itens de uma amostra de texto.

Da mesma forma, outras tarefas bastante utilizadas na etapa de pré-processamento por projetos é a transformação de letras maiúsculas em minúsculas ("*lowercase*"), para que haja uma uniformização das palavras do vocabulário, e a remoção de conteúdo XML (conteúdos de tags) e possíveis e-mails que podem existir no texto.

2.4. Trabalhos Correlatos

Cirqueira et al. (2018) apresenta em seu artigo uma revisão da literatura focada em buscar e analisar os métodos e técnicas principais utilizados na etapa de pré-processamento de dados em projetos voltados para de mídias sociais para Análise de Sentimentos. A pesquisa faz uma varredura em projetos que possuem seu escopo na língua portuguesa do Brasil, dada a riqueza e variedade linguística apresentada por este país. Assim, os autores conseguem

abstrair as principais implementações em Pré-Processamento utilizadas, construindo tabelas e gráficos indicativos da análise, contribuindo para o presente trabalho com uma forma estruturada para apresentação da análise de diferentes tarefas de processamento. Os autores chegam à conclusão de que não há uma padronização de framework de pré-processamento seguida pela comunidade desenvolvedora, onde, de maneira geral, cada estudo apresenta passos específicos.

O estudo comparativo de Grancharova e Jangefalk (2018) explora algoritmos de aprendizagem e procura investigar como a métrica de acurácia em modelos de classificação pode ser afetada por diferentes algoritmos e técnicas de pré-processamento para dados específicos, analizando a performance combinada destes algoritmos. Os autores constroem e executam diversos testes, utilizando combinações de tarefas de pré-processamento e algoritmos de classificação, anotando e comparando os resultados obtidos. Para os testes realizados, a conclusão revela que a permutação de remoção de *stopwords* e corretor ortográfico aumenta a acurácia para todos os classificadores estudados. A pesquisa construída pelos autores traz grandes benefícios demonstrando um fluxo a ser seguido para a construção dos testes e suas execuções, além de demonstrar que a acurácia consegue ser uma métrica eficaz para comparar resultados de classificadores.

Işik e Dağ (2019) buscam analisar as técnicas desenvolvidas em pré-processamento e suas influências em avaliações de revisões, além de tentar explicar observações e resultados interessantes sobre o desempenho de uma classificação de avaliações de revisões com cinco classes, contribuindo com uma estrutura de apresentação adequada para testes feitos em tarefas de pré-processamento com a incorporação de classificadores nas análises. A pesquisa conclui que eliminação de *stopwords* e palavras comuns, padronização de letras para minúsculas e combinações de 1 a 3 n-gramas possuem melhor performance do que outros métodos de pré-processamento para aumentar a acurácia em classificações. Os autores ainda discutem que os efeitos de abreviações, acrônimos, stemização e lematização podem ser maiores após a execução da transformação de letras para minúsculas.

A pesquisa de Chandrasekar e Qian (2019) possui o objetivo de estudar os impactos da aplicação de pré-processamento de dados na performance do classificador Naïve Bayes. Para isso, os autores propõem analisar esta etapa de preparação dos dados comparando os resultados obtidos, utilizando o classificador com o propósito de identificar e-mail que possuem conteúdo de spam. Com os testes realizados comparando a performance com e sem a utilização da etapa de preparação, os resultados obtidos favorecem o uso do pré-processamento adequado para tratamento dos dados. Os autores conseguem contribuir na

medida em que realizam comparações baseadas em performances sem a etapa de estudo, estabelecendo um ponto de partida para as análises.

A seguir são apresentadas algumas pesquisas elaboradas com o objeto da Mineração de Textos, com determinados trabalhos que foram desenvolvidos propriamente para o âmbito judiciário. A partir desse conjunto é possível verificar as principais tarefas empregadas na etapa de pré-processamento. Cada trabalho apresentado foi observado com foco em quais tarefas foram ou não exercidas, para possibilitar a formação posterior de uma tabela com estes dados. Com os dados, é possível estabelecer as melhores e mais eficazes tarefas de Pré-Processamento buscando alavancar os resultados das métricas.

Em sua dissertação, Andrade (2015) possuía o objetivo de abordar os algoritmos atuais para classificação em problemas com grande número de classes de documentos textuais. Além disso, a pesquisa propôs apontar um algoritmo de triagem de processos, realizando uma prova de conceito dada a triagem de denúncias recebidas pela Controladoria-Geral da União. Na etapa de Pré-Processamento é possível identificar diversas tarefas que o autor utilizou para preparar os elementos textuais. As tarefas são apresentadas na Tabela 1.

Com o objetivo da construção de uma aplicação da Inteligência Artificial (IA) na identificação de conexões pelo fato e tese jurídica em documentos de petições iniciais, Castro Júnior et al. (2020) propõe em seu artigo uma ferramenta que busque pelo aperfeiçoamento da gestão do conhecimento no judiciário, permitindo minerar petições iniciais de processos em tramitação, com processos distintos e petições iniciais idênticas em seus aspectos e com processos que possuem o mesmo fato e tese jurídica. No pré-processamento construído, utilizou as tarefas como a limpeza de caracteres especiais e *stopwords*, processo de stemização e tokenização.

Junior et al. (2018) apresenta em seu trabalho uma solução de extração de informações para predição dos resultados de sindicâncias utilizando a Mineração de Dados para automatizar e otimizar o acompanhamento destes processos. Como fonte da Base de Dados, a pesquisa utilizou como alvo o Diário Oficial de Pernambuco com documentos publicados de janeiro de 2007 e abril de 2017. Os autores utilizam as tarefas de remoção de *stopwords*, stemização e tokenização para realizar o Pré-Processamento dos dados.

Faraco (2020) demonstra em sua dissertação uma proposta de modelo de conhecimento que está direcionado para a extração de tópicos sobre documentos de julgados (acórdãos), como ponto de partida a análise de petições iniciais. A Base de Dados compõe documentos de acórdãos em formato JSON, capturados a partir de um crawler projetado para a Justiça Federal da 4ª Região, TRF4. Com o modelo construído a partir da aplicação de

Técnicas de Modelagem de Tópicos por meio do algoritmo de Análise Latente de Dirichlet (LDA) para apoiar o analista processual na análise das petições iniciais. Suas tarefas de Pré-Processamento aplicadas foram a retirada de caracteres especiais, números, acentuação, *stopwords*, também realizando a lematização e a tokenização.

Na dissertação apresentada por Sousa (2019) é possível identificar a utilização da Mineração de Textos, com o desenvolvimento de uma solução de apoio por meio da Inteligência Artificial com o intuito de aumentar a celeridade na classificação de processos eletrônicos, auxiliando no cadastro de petições iniciais e assuntos do processo. Neste projeto optou por utilizar um corpus de Petição Inicial em formato PDF e os classificadores Naïve Bayes, KNN, Árvore de Decisão, SVM, *K-Means* para estudo. A etapa de Pré-Processamento foi composta por remoção de números, pontuações e *stopwords*, transformar letras maiúsculas em minúsculas, utilizou também stemização e tokenização.

Mastella (2021) em sua dissertação utiliza a Mineração de Textos a partir da aplicação da modelagem de Classificação em textos jurídicos, propondo uma metodologia que otimize a escolha de parâmetros em algoritmos classificadores paralelizando o treinamento das técnicas com melhor desempenho, estudando as métricas geradas como resultado. Para isso, Pré-Processamento foi modelo utilizando tokenização, padronizando letras para minúsculas, exclusão de pontuações e *stopwords*.

A dissertação de Castro (2019) aborda a aplicação da Aprendizagem Profunda com o objetivo de identificar Entidades Nomeadas em documentos jurídicos. O autor concorda que a tarefa de aplicação de algoritmos em domínio judiciário recebe uma camada de complexidade superior por conta de seu léxico rico e particular. Para isso, são utilizadas as Redes Neurais Profundas para avaliar a representação das palavras para este domínio. Remoção de conteúdo XML, remoção de números, pontuação e caracteres especiais e o processo de tokenização foram tarefas realizadas no Pré-Processamento dos elementos textuais de treinamento.

Ferreira (2018) em sua monografia utiliza a Inteligência Artificial no âmbito jurídico com a Classificação de peças processuais de origem do Supremo Tribunal Federal (STF). O autor identificou que um dos problemas relacionados aos documentos seria que estes são advindos de diferentes fontes e assim não possuem padronização em códigos de identificadores de peças e marcadores de volumes, prejudicando a celeridade de processos. Logo, a pesquisa busca identificar automaticamente o corpo de diferentes peças processuais em um único documento. O Pré-Processamento utiliza-se das tarefas de remoção de números, espaços em branco, *stopwords* e *tags* XML, stemização e tokenização.

O artigo de Silva et al. (2021) realiza um estudo comparativo de abordagens utilizadas na Classificação de documentos, tomando como foco a problemática de classificação para documentos de vários tipos em língua portuguesa no contexto jurídico. No estudo, cinco metodologias foram avaliadas na tarefa de reconhecimento de onze tipos de petições do Tribunal de Justiça do Estado de Alagoas. A abordagem com maior destaque em desempenho foi utilizando a vetorização Frequência do Termo-Inverso da Frequência nos Documentos (*Term Frequency–Inverse Document Frequency*, TF-IDF) com classificador SVM. Os autores realizaram os processos de remoção de acentos, pontuações, espaços em branco, *stopwords*, padronização para minúsculo, stemização e tokenização.

Gusmão et al. (2021) investiga em seu artigo técnicas de Processamento de Linguagem Natural (PLN) em textos de Denúncias Criminais que possuem como origem o aplicativo do Disque Denúncia RJ. Neste caso, o objetivo é reduzir o tempo de análise das mensagens que possuem seu conteúdo escrito em português coloquial e muito informal possuindo, algumas vezes, erros morfológicos. Dessa forma, o trabalho utiliza o classificador SVM e busca identificar as melhores técnicas de Pré-Processamento visando alcançar a melhor acurácia. As tarefas utilizadas nesta etapa foram: remoção de acentos, dígitos, pontuação, *stopwords*, padronização em minúsculas, tokenização e stemização.

A plataforma Sinapses/CNJ surge no âmbito jurídico, como descrevem Pereira e Rodrigues (2022), como um sistema para armazenamento e distribuição de modelos de Inteligência Artificial construídos por entidades que objetivam alavancar as tarefas do poder judiciário. Esta plataforma, criada pela equipe de desenvolvimento do Tribunal de Justiça do Estado de Rondônia, atua como uma unificação de modelos de IA que podem ser utilizados por diversos órgãos jurídicos na esfera nacional. O Sinapses/CNJ possui algumas bibliotecas próprias escritas na linguagem Python que estão disponíveis para utilização pelos desenvolvedores. Uma destas, a biblioteca ia_utils, contém um módulo de normalizador de textos e sua versão 1.4.1, criada em 23 de fevereiro de 2018 abrange algumas tarefas de Pré-Processamento, que são: remoção de espaçamentos, generalização de termos, transformação de números para a forma escrita por extenso, padronização para minúsculas.

Para a elaboração da Tabela 1 foram verificadas as tarefas exercidas pelos estudos na etapa de Pré-Processamento, seguindo a premissa de marcar apenas as tarefas que o estudo descreveu em sua pesquisa. Logo, para casos em que o estudo não deixa clara a utilização da tarefa, esta não consta na tabela. Para apontar as tarefas que cada pesquisa realizou, foram empregadas seções com tarefas que possuem objetivos similares ou fazem parte de um mesmo escopo, como por exemplo as tarefas de Stemização e Lematização.

Nesse contexto, as tarefas dividem-se em:

- T1 - Remoção de Números e/ou Dígitos;
- T2 - Remoção de Espaços em Branco;
- T3 - Remoção de Pontuação, Acentuação e/ou caracteres especiais;
- T4 - Transformação de letras maiúsculas em minúsculas;
- T5 - Remoção de stopwords;
- T6 - Stemização e/ou Lematização;
- T7 - Tokenização; T8 - Remoção de conteúdo XML (tags) e/ou e-mails.

Neste cenário, a Tabela 1 apresenta os estudos relacionados com as tarefas encontradas.

Tabela 1. Técnicas de Pré-Processamento Implementadas nas Pesquisa.

Artigo \ Tarefa	T1	T2	T3	T4	T5	T6	T7	T8
Andrade (2015)	X	X	X	X	X	X		
Castro Júnior et al. (2020)			X	X	X	X	X	
Junior et al. (2018)					X	X	X	
Faraco (2020)	X		X		X	X	X	
Sousa (2019)	X		X	X	X	X	X	
Mastella (2021)			X	X	X		X	
Castro (2019)	X		X				X	X
Ferreira (2018)	X	X	X		X	X	X	X
Silva et al. (2021)		X	X	X	X	X		
Gusmão (2021)	X		X	X	X	X	X	
Normalizador da Plataforma Sinapses/CNJ	*	X		X		*		
TOTAL	7	4	9	7	9	9	8	2

Fonte: Autor (2023).

A marcação (X) representa que o trabalho desenvolvido pelos autores implementa aquele tipo de tarefa de Pré-Processamento. Para o Normalizador da Plataforma Sinapses/CNJ

as tarefas T1 e T6 possuem algumas observações (*), pois não empregam a técnica da forma como é descrita.

A tarefa T1 (Remoção de Números e/ou Dígitos) não é implementada pelo normalizador, porém existe uma conversão de números para a forma escrita por extenso. Já a tarefa T6 (Stemização/Lematização) também não é incorporada, mas há uma forma de generalização de termos nos textos processados.

3. FRAMEWORK EXPERIMENTAL

Esta seção compreende o desenvolvimento do projeto guiado pelas pesquisas realizadas. Como descrito nos objetivos, um fluxo de Pré-Processamento para Textos Jurídicos foi desenvolvido e, dessa forma, a seguir são apresentados os principais passos executados para alcançar os objetivos.

Para a construção e execução de todos os códigos da análise foi utilizada a linguagem de programação Python em sua versão 3.6.8, contendo bibliotecas adequadas para utilização e aplicação de todos as técnicas de pré-processamento analisadas. O fluxo projetado emprega as bibliotecas a partir da criação de funções próprias, sendo posteriormente utilizadas para os testes descritos a seguir.

Para a construção deste fluxo é necessário inicialmente projetar um fluxo de atividades correspondente às etapas do processo de avaliação e desenvolvimento. Com a incorporação das técnicas de Pré-Processamento analisadas a partir dos Estudos de Caso descritos anteriormente, uma Base de Dados com elementos de documentos jurídicos será aplicada no processo de Classificação.

Cada atividade a ser realizada que compõem a estrutura de análise e avaliação das tarefas será descrita abaixo. Antes de aplicar os dados nos testes e anotar os resultados torna-se essencial preparar a estrutura de cada teste, garantindo que exista uma padronização na execução e anotação para que a fase de comparação se torne mais eficaz.

Para seguir melhor descrição e estruturação, os testes seguiram níveis de incorporação das técnicas, definidos em 0 (sem utilização de pré-processamento), 1 (utilizar técnicas individualmente) e 2 (combinar técnicas).

3.1. Descrição da Base de Dados e Preparação dos Testes

A fase inicial consiste em abstrair os passos seguintes, verificando o que deve ser preparado anteriormente para que cada uma das etapas seja executada da melhor forma. Para isso, o primeiro ponto a ser discutido é a Base de Dados a ser utilizada, que se fará presente em todas as fases dos testes.

Uma base com elementos bem estruturados e definidos deve ser empregada para garantir a consistência das execuções. Os experimentos realizados utilizaram uma Base de Dados constituída de documentos jurídicos, sendo uma base de Petições Iniciais com 3000 elementos fornecida pelo Tribunal de Justiça do Maranhão, como parte da parceria entre o

TJMA e a Universidade Estadual do Maranhão. A base de dados é composta por cinco classes de petições: "IRDR1", "RR986", "RR1", "RR1061", "RR1074". Dada a execução dos testes, é possível então utilizar duas Base de Dados com elementos de tipos diferentes e submeter estas aos testes de maneira igual.

O próximo ponto a ser abordado sobre a fase de preparação é a definição de quais técnicas de Pré-Processamento serão implementadas e incorporadas nos testes, dados os níveis 0, 1 e 2 de desenvolvimento. As tarefas discutidas e apresentadas anteriormente foram: Remoção de Números e/ou Dígitos; Remoção de Espaços em Branco; Remoção de Pontuação, Acentuação e/ou caracteres especiais; Transformação de letras maiúsculas em minúsculas; Remoção de *stopwords*; Stemização e/ou Lematização; Tokenização; e Remoção de conteúdo XML (*tags*) e/ou e-mails.

Porém, nem todas estas tarefas serão testadas e incluídas no fluxo de processamento final por não ser possível realizar todos os testes satisfatoriamente em tempo hábil. Dessa forma, serão escolhidas as técnicas que foram mais utilizadas pelas pesquisas nos Estudos de Caso apresentados. Todos os testes a serem realizados serão documentados e descritos de forma a ser conhecidas quais técnicas foram utilizadas em cada nível de teste.

Da mesma maneira, é necessário entender e apontar quais algoritmos serão aplicados na fase de Mineração de Dados. A Classificação se torna uma escolha interessante por conta de ser uma técnica de Aprendizado de Máquina Supervisionado, garantindo que as classes e rótulos dos dados sejam previamente conhecidos, tomando como base os documentos de Petições Iniciais.

Para realizar a vetorização dos dados textuais dos documentos foi utilizado o modelo TF-IDF (*Term Frequency — Inverse Data Frequency*), convertendo assim os dados para o formato numérico antes de serem repassados para os classificadores. A classificação foi escolhida como algoritmo alvo para os testes, extraindo a acurácia de três classificadores diferentes: SVM (*Support Vector Machine*, ou "Máquina de Vetores de Suporte"), KNN (K-*Nearest Neighbors*, ou "K-Vizinhos Mais Próximos") e LR (*Logistic Regression*, ou "Regressão Logística"). Foi utilizada a técnica de validação cruzada *k-fold* para avaliação dos modelos com k igual a 10, gerando assim 10 acurácias diferentes para cada classificador, sendo registradas as acurácias para o melhor caso e a média entre as 10 obtidas.

3.2. Performance sem Pré-Processamento

O Nível 0 de avaliação se dá por apontar os resultados iniciais de métricas para as comparações. É o ponto de partida para conhecer como os classificadores atuam nas Base de Dados passadas. Anotar estes resultados será a função principal desta fase da avaliação. Logo, os passos principais a serem realizados no Nível 0 são: Aplicar os Classificadores e Anotar as Métricas.

Com a Base de Dados e os classificadores previamente definidos na Fase de Preparação, estes serão testados no Nível 0 para compor os primeiros resultados anotados para as métricas de Acurácia, podendo ser representados por tabelas e gráficos construídos ao final de todas as fases de desenvolvimento.

A importância de gerar estas métricas com a classificação sem executar a etapa de Pré-Processamento está em obter medidas regulares do comportamento para a estrutura definida e incluir esta etapa de maneira concisa.

3.3. Técnicas Individuais

O próximo nível de testes é o Nível 1, compondo a execução da mesma estrutura mencionada anteriormente que será executada no Nível 0, porém com a adição da etapa de Pré-Processamento. Dessa forma, esta fase de testes iniciará o entendimento de quais são as principais técnicas a serem exploradas na composição de uma estrutura de pré-processamento voltada para Textos Jurídicos. A sequência de passos no Nível 1 de desenvolvimento segue: Aplicar o Pré-Processamento Individualmente, Aplicar os Classificadores e Anotar as Métricas.

O passo de Aplicar o Pré-Processamento Individual é o centro desta fase de testes, onde cada técnica será estudada de singular, sem adicionar ou mesclar nenhuma das técnicas. Com a aplicação dos classificadores é possível anotar os resultados e avaliar o acréscimo ou decréscimo das métricas com relação aos resultados obtidos no Nível 0 de desenvolvimento. Como descrito anteriormente, não serão aplicadas todas as tarefas de Pré-Processamento, mas somente as mais utilizadas pelas pesquisas realizadas.

3.4. Combinações de Técnicas

Seguindo com a realização dos testes, o Nível 2 de desenvolvimento consiste em descrever e apropriar a avaliação de combinações das melhores técnicas estudadas e testadas

até o Nível 1. O nível atual segue um fluxo de passos similar ao anterior, com a principal diferença na primeira parte. Os passos a serem realizados são: Aplicar o Pré-Processamento Combinado, Aplicar os Classificadores e Anotar as Métricas.

Assim como na fase anterior, o ponto central que irá diferenciar os resultados da fase atual é a forma como o Pré-Processamento é aplicado. Neste caso, serão tomadas algumas das técnicas utilizadas no Nível 1 e colocadas em um fluxo de processos, compondo um pipeline de execução para seguir com os resultados gerados pelos classificadores adotados.

Inicialmente, são tomadas tarefas de duas em duas, anotando as acurácias encontradas. Então são tomadas tarefas de três em três para a combinação. Após isso, para avançar nas combinações foram definidos alguns fatores que os as combinações precisam cumprir para avançar para a próxima etapa. As regras que a combinação precisa cumprir estão descritas na Tabela 2.

A regra 1 é definida como "A acurácia da combinação deve ser maior que a acurácia das tarefas combinadas". A regra 2 representa "A acurácia da combinação deve ser maior que a média das acurácias das tarefas combinadas". Já a regra segue como "A acurácia da combinação deve ser maior que a acurácia máxima entre as tarefas combinadas".

Tabela 2. Descrição das Regras Utilizadas.

	Regra	**Exemplo para Combinação de 3**
Regra 1	Cx/.../z > (Tx,...,Tz)	Cx/y/z > (Tx,Ty,Tz)
Regra 2	Cx/.../z > MED(Tx,...,Tz)	Cx/y/z > MED(Tx,Ty,Tz)
Regra 3	Cx/.../z > MAX(Tx,...,Tz)	Cx/y/z > MAX(Tx,Ty,Tz)

Fonte: Autor (2023).

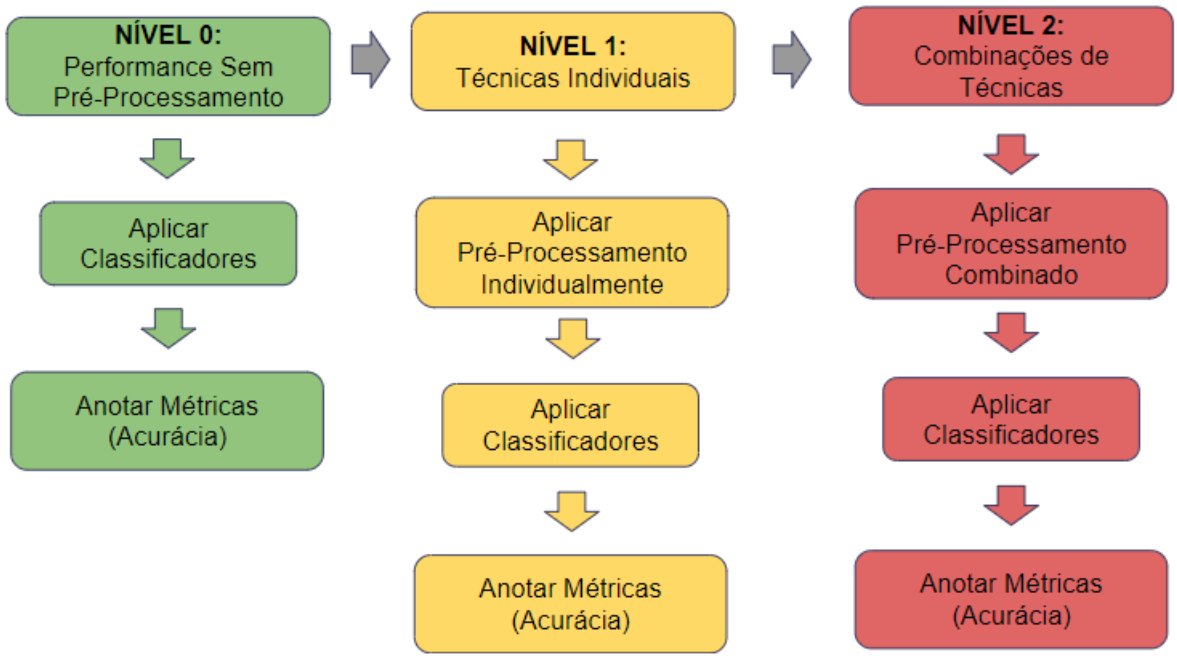
3.5. Comparação de Resultados e Construção de Algoritmos

A última fase que compõe o fluxo de avaliação das técnicas possui o objetivo de alinhar todos os testes realizados, abstraindo por meio de tabelas os resultados gerados pelas etapas anteriores com as métricas de Acurácia dos classificadores a partir das Bases e técnicas utilizadas.

A organização dos dados obtidos se dará a depender de quais parâmetros foram empregados, porém o máximo de informação será agregada para que se obtenha o melhor entendimento e seja possível realizar as comparações dos resultados. Após esta análise é possível construir uma codificação com as técnicas combinadas que apresentaram as melhores métricas para os testes realizados.

Abaixo a Figura 4 apresenta sucintamente os níveis que irão compor os testes para análise.

Figura 4. Níveis de Execução do Framework Experimental.



Fonte: Autor (2023).

4. RESULTADOS

Segundo a arquitetura descrita, foram elaborados e executados os testes para as técnicas de forma individual e as combinações de tarefas. Para melhor apresentação, as tabelas seguem a descrição TX (Tarefa X), sendo X o número da tarefa, com a seguinte ordem apresentada na Tabela 3.

Tabela 3. Descrição das Tarefas Utilizadas.

Tarefa 1 (T1)	Remoção de Números/Dígitos
Tarefa 2 (T2)	Remoção de *links* e emails
Tarefa 3 (T3)	Remoção de Espaços em Branco
Tarefa 4 (T4)	Remoção de *stopwords*
Tarefa 5 (T5)	Transformação de maiúsculas em minúsculas
Tarefa 6 (T6)	Remoção de Pontuação, Acentuação e caracteres especiais
Tarefa 7 (T7)	Stemização
Tarefa 8 (T8)	Lematização
Tarefa 9 (T9)	Tokenização

Fonte: Autor (2023).

A Tabela 4 apresenta as acurácias médias encontradas para os Classificadores sem a utilização de técnicas de Pré-Processamento (Nível 0) e para as Técnicas Individuais (Nível 1). Os valores destacados representam as acurácias das Técnicas Individuais (Nível 1) que ultrapassaram os valores Sem Pré-Processamento (*Baseline*) para cada Classificador.

Tabela 4. Performances dos Níveis 0 e 1 - Acurácias Médias.

	Regressão Logística		**KNN**		**SVM**	
	Acurácia Média (%)	**Desvio Padrão**	**Acurácia Média (%)**	**Desvio Padrão**	**Acurácia Média (%)**	**Desvio Padrão**
Baseline	73,83	0,03652	72,66	0,04353	68,33	0,03821
Tarefa 1	73,80	0,03773	**73,09**	0.03796	**68,61**	0,03907

Tarefa 2	73,63	0,03775	72,64	0,04247	68,04	0,03604
Tarefa 3	73,83	0,03652	72,66	0,04353	68,33	0,03821
Tarefa 4	72,24	0,03529	72,39	0,03812	68,07	0,03830
Tarefa 5	73,83	0,03652	72,66	0,04353	68,33	0,03821
Tarefa 6	**74,02**	0,03587	72,61	0,04217	**68,43**	0,03757
Tarefa 7	**74,04**	0,03945	**73,89**	0,04379	**68,82**	0,03859
Tarefa 8	73,73	0,03524	**73,20**	0,03725	**68,59**	0,03889
Tarefa 9	**73,98**	0,03879	**73,85**	0,04118	**68,53**	0,03725

Fonte: Autor (2023).

A Tabela 5 apresenta as acurácias de melhor caso encontradas para os classificadores sem a utilização de técnicas de Pré-Processamento (Nível 0) e para as Técnicas Individuais (Nível 1). Os valores destacados representam as acurácias das Técnicas Individuais (Nível 1) que ultrapassaram os valores base sem Pré-Processamento para cada Classificador.

Tabela 5. Performances dos Níveis 0 e 1 - Acurácias de Melhor Caso.

	Regressão Logística	**KNN**	**SVM**
Sem pré-processamento	82,13	79,24	74,95
Tarefa 1	81,79	**80,44**	74,95
Tarefa 2	**82,48**	**79,71**	74,95
Tarefa 3	82,13	79,24	74,95

Tarefa 4	79,24	**79,76**	74,95
Tarefa 5	82,13	79,24	74,95
Tarefa 6	**84,56**	**81,22**	74,95
Tarefa 7	**83,30**	**80,12**	**75,45**
Tarefa 8	79,58	**81,03**	74,60
Tarefa 9	**82,38**	**82,14**	74,24

Fonte: Autor (2023).

As Tabelas 4 e 5 apresentam os resultados que irão servir de base para as próximas execuções. É possível analisar pela Tabela 4 que, para os resultados de Acurácia Média, o Classificador SVM apresenta cinco tarefas com resultados que ultrapassam a acurácia de Baseline, seguido pelo Classificador KNN, com quatro tarefas, enquanto a Regressão Logística apresenta apenas três tarefas que ultrapassam o Baseline. Por outro lado, a Tabela 5 demonstra como o classificador KNN supera os demais, possuindo sete resultados de acurácia que ultrapassam o Baseline, enquanto a Regressão Logística e o SVM apresentam, respectivamente, quatro acurácias e uma acurácia que se sobressaem.

Seguindo para o Nível 2, foi iniciada a etapa de combinação das tarefas. Primeiramente, foram definidas combinações agregando de duas em duas tarefas, como por exemplo: Remoção de Números/Dígitos e Remoção de links e *emails*. Ao todo, foram obtidos 72 resultados para esta etapa, combinando as 9 tarefas entre si sem que uma mesma tarefa fosse repetida na combinação.

Assim, a Tabela 6 apresenta as acurácias médias encontradas para a combinação de duas técnicas utilizando como classificador a Regressão Logística. As linhas representam qual tarefa foi executada primeiro e as colunas representam a tarefa que foi executada por último.

Tabela 6. Combinação de 2 tarefas - Regressão Logística (Acurácia Média).

	T1 (%)	**T2 (%)**	**T3 (%)**	**T4 (%)**	**T5 (%)**	**T6 (%)**	**T7 (%)**	**T8 (%)**	**T9 (%)**
T1	X	71.789	72.578	72.049	73.807	72.578	72.662	72.178	72.413
T2	73.667	X	72.578	71.789	73.667	72.578	72.662	72.178	72.413
T3	73.807	71.789	X	72.049	73.807	72.578	72.662	72.178	72.413
T4	72.003	71.789	72.558	X	72.049	72.590	72.662	72.303	72.134
T5	73.807	71.789	72.578	72.049	X	72.578	72.662	72.178	72.413
T6	**74.190**	71.603	72.556	71.552	**74.190**	X	72.316	72.556	72.671
T7	**74.139**	**74.041**	73.772	73.816	**74.314**	**74.270**	X	**74.048**	**73.888**
T8	73.377	73.500	73.775	73.322	73775	**73.854**	**74.039**	X	72.447
T9	**74.049**	72.754	72.413	72.982	**74.049**	72.413	73.418	71.923	X

Fonte: Autor (2023).

A Tabela 7 apresenta as acurácias de melhor caso encontradas para a combinação de duas técnicas utilizando como classificador a Regressão Logística. As linhas representam qual tarefa foi executada primeiro e as colunas representam a tarefa que foi executada por último.

Tabela 7. Combinação de 2 tarefas - Regressão Logística (Acurácia Melhor Caso).

	T1 (%)	**T2 (%)**	**T3 (%)**	**T4 (%)**	**T5 (%)**	**T6 (%)**	**T7 (%)**	**T8 (%)**	**T9 (%)**
T1	X	78.079	79.375	78.079	81.791	79.375	79.763	79.608	78.680
T2	80.162	X	79.375	78.079	80.162	79.375	79.763	79.608	78.680
T3	81.791	78.079	X	78.079	81.791	79.375	79.763	79.608	78.680
T4	78.079	78.079	81.004	X	78.079	79.027	79.763	79.722	79.142
T5	81.791	78.079	79.375	78.079	X	79.375	79.763	79.608	78.680
T6	80.402	79.817	81.791	79.817	80.402	X	79.962	81.791	81.919
T7	**82.139**	**82.393**	78.989	**82.393**	**82.267**	81.698	X	**82.393**	80.977
T8	79.815	**82.139**	**82.486**	79.400	**82486**	**84.001**	81.199	X	80.288
T9	**82.498**	78.785	78.680	80.300	**82.498**	78.680	78.314	77.974	X

Fonte: Autor (2023).

As Tabelas 6 e 7 compõem os resultados de acurácias quando aplicado o Classificador de Regressão Logística. Percebe-se, pelos resultados, que, para Acurácia Média, apenas doze das acurácias se sobressaem em relação à acurácia de Baseline, totalizando 16%. Para Acurácia de Melhor Caso, somente 11 resultados conseguem ultrapassar o Baseline, assim formando 15,2%. Dos 144 resultados de acurácia, somente 15,97%, 23 acurácias, obtiveram resultado positivo.

A Tabela 8 apresenta as acurácias médias encontradas para a combinação de duas técnicas utilizando como classificador o KNN. As linhas representam qual tarefa foi executada primeiro e as colunas representam a tarefa que foi executada por último.

Tabela 8. Combinação de 2 tarefas - KNN (Acurácia Média).

	T1 (%)	**T2 (%)**	**T3 (%)**	**T4 (%)**	**T5 (%)**	**T6 (%)**	**T7 (%)**	**T8 (%)**	**T9 (%)**
T1	X	72.557	72.563	72.545	**73.090**	72.563	**73.774**	**73.092**	**73.439**
T2	**73.520**	X	72.563	72.557	**73.520**	72.563	**73.774**	**73.092**	**73.439**
T3	**73.090**	72.557	X	72.545	**73.090**	72.563	**73.774**	**73.092**	**73.439**
T4	72.537	72.557	72.554	X	72.545	72.410	**73.774**	**72.459**	**73.500**
T5	**73.090**	72.557	72.563	72.545	X	72.563	**73.774**	**73.092**	**73.439**
T6	72.001	71.442	**72.942**	71.442	72.001	X	72.117	72.942	**73.565**
T7	**74.215**	**73.815**	**72.838**	**73.315**	**73.989**	**73.077**	X	**74.008**	**73.053**
T8	**74.221**	**73.627**	**73.584**	72.582	**73.584**	72.467	**73.188**	X	**73.286**
T9	**73.418**	**73.857**	**73.439**	**73.664**	**73.418**	**73.439**	**73.910**	72.300	X

Fonte: Autor (2023).

A Tabela 9 apresenta as acurácias de melhor caso encontradas para a combinação de duas técnicas utilizando como classificador o KNN. As linhas representam qual tarefa foi executada primeiro e as colunas representam a tarefa que foi executada por último.

Tabela 9. Combinação de 2 tarefas - KNN (Acurácia Melhor Caso).

	T1 (%)	T2 (%)	T3 (%)	T4 (%)	T5 (%)	T6 (%)	T7 (%)	T8 (%)	T9 (%)
T1	X	**81.910**	**81.094**	**82.024**	**80.444**	**81.094**	**80.166**	**81.094**	**79.350**
T2	**80.444**	X	**81.094**	**81.910**	**80.444**	**81.094**	**80.166**	**81.094**	**79.350**
T3	**80.444**	**81.910**	X	**82.024**	**80.444**	**81.094**	**80.166**	**81.094**	**79.350**
T4	**82.024**	**81.910**	78.534	X	**82.024**	79.002	**80.166**	78.996	78.438
T5	**80.444**	**81.910**	**81.094**	**82.024**	X	**81.094**	**80.166**	**81.094**	**79.350**
T6	**81.612**	78.647	**83.241**	78.647	**81.612**	X	**82.073**	**83.241**	**82.085**
T7	**81.857**	**80.630**	78.994	**80.050**	**81.624**	**80.048**	X	**81.091**	**81.449**
T8	**82.779**	**81.458**	**81.805**	76.912	**81.805**	78.207	**79.654**	X	**83.075**
T9	**81.964**	**80.066**	**79.350**	**80.079**	**81.964**	**79.350**	**81.810**	77.124	X

Fonte: Autor (2023).

As Tabelas 8 e 9 compõem os resultados de acurácias ao aplicar KNN como Classificador. É possível identificar que, para Acurácia Média, quarenta e quatro das acurácias se sobressaem em relação à acurácia de Baseline, totalizando 61,11% das setenta e duas acurácias totais. Para Acurácia de Melhor Caso, sessenta e dois resultados conseguem ultrapassar o Baseline, apresentando 86,11%. Assim, dos 144 resultados de acurácia, um total de 73,61%, 106 acurácias, obtiveram resultado positivo.

A Tabela 10 apresenta as acurácias médias encontradas para a combinação de duas técnicas utilizando como classificador o SVM. As linhas representam qual tarefa foi executada primeiro e as colunas representam a tarefa que foi executada por último.

Tabela 10. Combinação de 2 tarefas - SVM (Acurácia Média).

	T1 (%)	T2 (%)	T3 (%)	T4 (%)	T5 (%)	T6 (%)	T7 (%)	T8 (%)	T9 (%)
T1	X	68.088	68.238	68.111	**68.615**	68.238	68.053	68.153	67.997
T2	68.254	X	68.238	68.088	68.254	68.238	68.053	68.153	67.997
T3	**68.615**	68.088	X	68.111	**68.615**	68.238	68.053	68.153	67.997
T4	68.111	68.088	68.106	X	68.111	68.184	68.053	68.107	67.982

T5	**68.615**	68.088	68.238	68.111	X	68.238	68.053	68.153	67.997
T6	**68.642**	68.056	**68.847**	68.045	**68.642**	X	**68.525**	**68.847**	**68.363**
T7	**68.863**	67.915	68.020	67.889	**68.936**	68.075	X	68.141	67.967
T8	69.221	**68.859**	**69.208**	**68.358**	69.208	**69.223**	**69.157**	X	**69.244**
T9	**68.822**	**68.608**	67.997	**68.713**	**68.822**	67.997	**68.790**	**69.038**	X

Fonte: Autor (2023).

A Tabela 11 apresenta as acurácias de melhor caso encontradas para a combinação de duas técnicas utilizando como classificador o SVM. As linhas representam qual tarefa foi executada primeiro e as colunas representam a tarefa que foi executada por último.

Tabela 11. Combinação de 2 tarefas - SVM (Acurácia Melhor Caso).

	T1 (%)	**T2 (%)**	**T3 (%)**	**T4 (%)**	**T5 (%)**	**T6 (%)**	**T7 (%)**	**T8 (%)**	**T9 (%)**
T1	X	74.958	74.152	74.958	74.958	74.152	**75.313**	73.443	74.152
T2	74.958	X	74.152	74.958	74.958	74.152	**75.313**	73.443	74.152
T3	74.958	74.958	X	74.958	74.958	74.152	**75.313**	73.443	74.152
T4	74.958	74.958	74.152	X	74.958	74.152	**75.313**	73.798	74.152
T5	74.958	74.958	74.152	74.958	X	74.152	**75.313**	73.443	74.152
T6	74.958	74.958	74.958	74.958	74.958	X	74.958	74.958	**75.313**
T7	74.958	74.053	74.179	73.798	**75.313**	73.798	X	73.798	74.152
T8	74.958	74.958	75.454	73.443	**75.454**	**75.454**	**75.340**	X	**75.454**
T9	74.604	74.604	74.152	74.604	74.604	74.152	74.958	**75.794**	X

Fonte: Autor (2023).

As Tabelas 10 e 11 apresentam os resultados de acurácias para o Classificador SVM. De acordo com os resultados, é possível identificar que, para os resultados de Acurácia Média, apenas vinte e quatro das acurácias se sobressaem em relação à acurácia de Baseline, totalizando 33,33%. Para Acurácia de Melhor Caso, somente 12 resultados conseguem ultrapassar o Baseline, assim compondo 16,66%. Do total de 144 resultados de acurácia, somente 25%, 36 acurácias, obtiveram resultado positivo.

É possível identificar que, para alguns classificadores, as tarefas de Pré-Processamento executadas não revelaram nenhum ou quase nenhum ganho. Em alguns casos, os resultados foram até abaixo da acurácia base sem Pré-Processamento, que podemos chamar de acurácias negativas. As acurácias que foram acima da base podem ser chamadas de positivas.

Como observado na Tabela 9, o classificador KNN obteve **62** resultados positivos dos 72 totais para a acurácia no melhor caso, se sobressaindo muito em comparação com a Regressão Logística (apenas **11** resultados positivos no melhor caso) e o SVM (apenas **12** resultados positivos no melhor caso). Logo, todos os resultados encontrados para as etapas seguintes são acurácias encontradas com o classificador KNN. A seguir, a Tabela 12 demonstra a relação de acurácias positivas encontradas pelos classificadores, onde o KNN se sobressai tanto para as acurácias médias quanto para as acurácias no melhor caso.

Tabela 12. Resultados Positivos - Combinação de 2 tarefas.

	LR	**KNN**	**SVM**
Resultados Positivos para Acurácia Média	12/72	**44/72**	24/72
Resultados Positivos para Acurácia no Melhor Caso	11/72	**62/72**	12/72
Total de Acurácias Positivas	23	**106**	36
Porcentagem de Acurácias Positivas	15,97%	**73,61%**	25%

A partir disso, foi escolhido apenas um classificador para realizar os testes seguintes, sendo definido o KNN pelo desempenho apresentado, apresentando apenas a sua acurácia para o melhor caso da Validação Cruzada.

Com estes resultados, a próxima etapa para o Nível 2 foi realizar a combinação das tarefas tomando de três em três. Para que não fosse necessário realizar várias combinações trocando a ordem de cada tarefa na execução, foi definida uma ordem para as tarefas de acordo com sua relação na combinação de duas em duas.

Por exemplo, para as tarefas T1 e T2, quando se toma T1 anteriormente a T2, a acurácia encontrada é **81.910%**, já tomando T2 antes de T1 a acurácia se torna **80.444%**. Logo, percebe-se que tomando primeiramente T1 o resultado é mais relevante.

Com isso, a ordem passada para a combinação de três tarefas se deu de acordo com os resultados encontrados na combinação de duas. Dessa forma, tomando as 9 tarefas para combinar foram obtidos 84 resultados de acurácias, apresentados na Tabela 13.

A Tabela 13 apresenta as combinações, demonstrando quais tarefas foram utilizadas, seguidas das acurácias média (AMed), com descrição do desvio padrão (DP) e de melhor caso (AMax).

Tabela 13. Combinação de 3 tarefas.

Combinação	Tarefa 1	Tarefa 2	Tarefa 3	AMed % (DP)	AMax (%)
1	T1	T2	T3	73.425 (0.04032)	**81.835**
2	T1	T2	T4	73.575 (0.04268)	**82.139**
3	T1	T2	T5	73.575 (0.04268)	**82.139**
4	T1	T2	T6	72.630 (0.04158)	**80.971**
5	T1	T2	T7	72.581 (0.03240)	78.106
6	T1	T2	T8	72.495 (0.03763)	78.351
7	T1	T2	T9	72.495 (0.03763)	78.351
8	T1	T3	T4	72.495 (0.03763)	78.351
9	T1	T3	T5	72.495 (0.03763)	78.351
10	T1	T3	T6	72.495 (0.03763)	78.351
11	T1	T3	T7	72.634 (0.03926)	**79.603**
12	T1	T3	T8	73.502 (0.04061)	**80.315**
13	T1	T3	T9	73.502 (0.04061)	**80.315**
14	T1	T4	T5	73.502 (0.04061)	**80.315**
15	T1	T4	T6	73.502 (0.04061)	**80.315**
16	T1	T4	T7	72.860 (0.03981)	**80.883**
17	T1	T4	T8	72.526 (0.03731)	**81.231**
18	T1	T4	T9	71.980 (0.03899)	78.508
19	T1	T5	T6	71.512 (0.04073)	**80.402**
20	T1	T5	T7	71.860 (0.03578)	79.142

21	T1	T5	T8	72.055 (0.03727)	77.625
22	T1	T5	T9	72.055 (0.03727)	77.625
23	T1	T6	T7	72.333 (0.03931)	**79.256**
24	T1	T6	T8	72.832 (0.03949)	**80.201**
25	T1	T6	T9	72.832 (0.03949)	**80.201**
26	T1	T7	T8	71.868 (0.04084)	78.339
27	T1	T7	T9	73.727 (0.04196)	**81.476**
28	T1	T8	T9	73.727 (0.04196)	**81.476**
29	T2	T3	T4	73.764 (0.03652)	**81.141**
30	T2	T3	T5	73.764 (0.03652)	**81.141**
31	T2	T3	T6	73.073 (0.03405)	**79.985**
32	T2	T3	T7	73.679 (0.03439)	**79.715**
33	T2	T3	T8	73.145 (0.03488)	**80.404**
34	T2	T3	T9	73.534 (0.03235)	**79.375**
35	T2	T4	T5	72.991 (0.03680)	**80.290**
36	T2	T4	T6	72.991 (0.03680)	**80.290**
37	T2	T4	T7	71.109 (0.03951)	**79.740**
38	T2	T4	T8	72.554 (0.04172)	**80.087**
39	T2	T4	T9	72.485 (0.04111)	**80.201**
40	T2	T5	T6	72.554 (0.04172)	**80.087**
41	T2	T5	T7	72.320 (0.04123)	**80.434**
42	T2	T5	T8	72.917 (0.04276)	**80.434**
43	T2	T5	T9	72.938 (0.04056)	**80.087**
44	T2	T6	T7	72.945 (0.04216)	**80.434**
45	T2	T6	T8	73.126 (0.04219)	**80.434**
46	T2	T6	T9	72.918 (0.03913)	**80.087**

47	T2	T7	T8	73.079 (0.04188)	**80.434**
48	T2	T7	T9	73.684 (0.04389)	**83.541**
49	T2	T8	T9	73.242 (0.04201)	**83.888**
50	T3	T4	T5	73.683 (0.03848)	**82.411**
51	T3	T4	T6	73.450 (0.03946)	**80.895**
52	T3	T4	T7	72.947 (0.03844)	**81.128**
53	T3	T4	T8	72.735 (0.03438)	**79.740**
54	T3	T4	T9	72.894 (0.03646)	**79.740**
55	T3	T5	T6	72.777 (0.03403)	**79.740**
56	T3	T5	T7	71.332 (0.03693)	**80.087**
57	T3	T5	T8	73.032 (0.04101)	**80.087**
58	T3	T5	T9	73.098 (0.04054)	**80.127**
59	T3	T6	T7	73.240 (0.04041)	**80.087**
60	T3	T6	T8	73.295 (0.03931)	**80.087**
61	T3	T6	T9	73.054 (0.04024)	**80.087**
62	T3	T7	T8	72.884 (0.04039)	**80.087**
63	T3	T7	T9	72.857 (0.04079)	**80.087**
64	T3	T8	T9	72.993 (0.04106)	**80.087**
65	T4	T5	T6	73.450 (0.03946)	**80.895**
66	T4	T5	T7	72.947 (0.03844)	**81.128**
67	T4	T5	T8	72.735 (0.03438)	**79.740**
68	T4	T5	T9	72.894 (0.03646)	**79.740**
69	T4	T6	T7	71.332 (0.03693)	**80.087**
70	T4	T6	T8	72.873 (0.03826)	**80.087**
71	T4	T6	T9	73.377 (0.03931)	**80.127**

72	T4	T7	T8	73.483 (0.03948)	**80.087**
73	T4	T7	T9	73.770 (0.04117)	**80.482**
74	T4	T8	T9	73.527 (0.04120)	**80.127**
75	T5	T6	T7	73.747 (0.04326)	**81.136**
76	T5	T6	T8	72.931 (0.04154)	**80.094**
77	T5	T6	T9	73.657 (0.04358)	**81.835**
78	T5	T7	T8	72.102 (0.04075)	**79.507**
79	T5	T7	T9	73.063 (0.04525)	**81.136**
80	T5	T8	T9	72.803 (0.04495)	**81.128**
81	T6	T7	T8	72.931 (0.04154)	**80.094**
82	T6	T7	T9	72.906 (0.04324)	**81.136**
83	T6	T8	T9	73.013 (0.04424)	**81.022**
84	T7	T8	T9	73.563 (0.04532)	**85.518**

Fonte: Autor (2023).

A Tabela 14 apresenta os resultados para a combinação de três tarefas apenas para aqueles que satisfizeram todas as condições descritas na Tabela 2. Caso qualquer uma das regras não fossem cumpridas, a combinação seria descartada. Dos 84 resultados, apenas 8 resultados de acurácia conseguiram satisfazer as condições, eliminando combinações que não produzem ganho significativo. Também é possível perceber que algumas combinações alcançaram resultados muito elevados em comparação com a execução sem Pré-Processamento do KNN, como é o caso para a combinação entre T7, T8 e T9 com acurácia de **85.518%**. Os resultados destacados são aqueles que conseguiram ultrapassar as acurácias para a combinação apenas com duas tarefas.

Tabela 14. Combinações de 3 tarefas que satisfazem as condições.

Combinação	Tarefa 1	Tarefa 2	Tarefa 3	AMed % (DP)	AMax (%)
2	T1	T2	T4	73.575 (0.04268)	**82.139**
3	T1	T2	T5	73.575 (0.04268)	**82.139**
27	T1	T7	T9	73.727 (0.04196)	**81.476**
30	T2	T3	T5	73.764 (0.03652)	**81.141**
48	T2	T7	T9	73.684 (0.04389)	**83.541**
49	T2	T8	T9	73.242 (0.04201)	**83.888**
50	T3	T4	T5	73.683 (0.03848)	**82.411**
84	T7	T8	T9	73.563 (0.04532)	**85.518**

Fonte: Autor (2023).

Prosseguindo com os testes do Nível 2, a próxima etapa seria realizar combinações aumentando o número de tarefas combinadas para quatro, cinco e assim sucessivamente. Para encontrar quais seriam as próximas combinações mais prováveis de sucesso, foram escolhidas somente as tarefas que aparecem na Tabela 14.

Assim, foram mescladas todas as linhas da Tabela 14 para encontrar as combinações adequadas. Por exemplo, ao mesclar a linha 1 (T1, T2 e T4) com a linha 2 (T1, T2 e T5) obtemos uma combinação de quatro tarefas (T1, T2, T4 e T5). Mesclando a linha 1 (T1, T2 e T4) com a linha 3 (T1, T7 e T9) obtemos uma combinação de cinco tarefas (T1, T2, T4, T7 e T9). Com isso, todas as combinações apresentadas nas Tabelas 15, 16 e 17 com suas acurácias foram elaboradas seguindo a regra mencionada anteriormente.

Tabela 15. Combinações de 4 tarefas.

Tarefas Combinadas	Acurácia Média % (DP)	Acurácia Melhor Caso (%)
T1, T2, T4, T5	73.575 (0.04268)	**82.139**
T1,T2, T5, T3	73.575 (0.04268)	**82.139**
T1, T7, T9, T2	72.875 (0.04260)	**81.225**

T1, T7, T9, T8	72.076 (0.03953)	**80.763**
T2, T3, T5, T4	71.994 (0.04013)	78.088
T2, T7, T9, T8	71.872 (0.04356)	**81.344**

Fonte: Autor (2023).

Tabela 16. Combinações de 5 tarefas.

Tarefas Combinadas	Acurácia Média % (DP)	Acurácia (%)
T1, T2, T4, T7, T9	71.896 (0.04349)	**81.344**
T1, T2, T4, T3, T5	71.877 (0.04138)	78.079
T1, T2, T4, T8, T9	72.082 (0.04594)	**83.206**
T1, T2, T5, T7, T9	72.031 (0.04474)	**81.691**
T1, T2, T5, T8, T9	72.155 (0.04428)	**81.691**
T1, T2, T5, T3, T4	71.928 (0.04148)	78.426
T1, T7, T9, T2, T8	71.965 (0.04417)	**81.691**
T2, T3, T5, T7, T9	71.965 (0.04417)	**81.691**
T2, T3, T5, T8, T9	71.996 (0.04400)	**81.577**

Fonte: Autor (2023).

Tabela 17. Combinações de 6 tarefas.

Tarefas Combinadas	Acurácia Média % (DP)	Acurácia (%)
T1, T2, T4, T7, T8, T9	71.946 (0.04413)	**81.577**
T1, T2, T5, T7, T8, T9	71.946 (0.04413)	**81.577**
T1, T7, T9, T2, T3, T5	71.946 (0.04413)	**81.577**
T1, T7, T9, T3, T4, T5	71.946 (0.04413)	**81.577**
T2, T3, T5, T7, T8, T9	71.946 (0.04413)	**81.577**
T2, T7, T9, T3, T4, T5	71.946 (0.04413)	**81.577**
T2, T8, T9, T3, T4, T5	71.969 (0.04418)	**81.577**
T3, T4, T5, T7, T8, T9	71.946 (0.04413)	**81.577**

Fonte: Autor (2023).

A Tabela 18 apresenta o resultado para a combinação que satisfez todas as condições descritas na Tabela 2. Como pode ser observado, apenas a combinação de T1, T2, T5, T7, T8 e T9 cumpriu as condições. Assim, como não há mais nenhuma forma de combinação que adicione novas tarefas, pode-se atribuir esta combinação como sendo a que possui maior quantidade de tarefas e que alcançou maior acurácia, dadas as tarefas executadas e a base de dados empregada.

Tabela 18. Combinações de 6 tarefas que satisfaz as condições.

Tarefas Combinadas	Acurácia Média % (DP)	Acurácia (%)
T1, T2, T5, T7, T8, T9	71.946 (0.04413)	**81.577**

Fonte: Autor (2023).

Abaixo serão apresentadas comparações dos resultados obtidos com o Estado da Arte em Pré-Processamento para Mineração de Textos a partir da apresentação e confronto com o Método Proposto da biblioteca LegalNLP.

4.1. Comparação do Método Proposto com Biblioteca LegalNLP

A biblioteca de Processamento de Linguagem Natural para a Linguagem Jurídica Brasileira, LegalNLP, como descreve Polo et al. (2021), disponibiliza funções que podem facilitar a manipulação de textos jurídicos em português, contendo também modelos de linguagem pré-treinados para a linguagem jurídica brasileira. Também possui demonstrações e tutoriais para auxiliar os usuários da ferramenta.

O LegalNLP apresenta e disponibiliza modelos de linguagem pré-treinados, funções e demonstrações específicas para a linguagem e área jurídica brasileira. O objetivo principal é catalisar o uso de ferramentas de processamento de linguagem natural para análise de textos legais pela indústria, governo e academia brasileira, fornecendo assim ferramentas necessárias e material de apoio acessível.

Polo et al. (2021) inclui dentro dessas funções disponíveis na biblioteca existem funções específicas para limpeza de texto, utilizando abordagens diferentes: Função “Clean” e Função “*Clean_BERT*” . A primeira abordagem, Função “*Clean*” se trata de uma função de limpeza de textos a ser utilizada em conjunto com os modelos *Doc2Vec*, *Word2Vec* e *FastText*, também é utilizado RegEx para mascarar e extrair informações como endereços de *e-mail*,

URLs, datas, números, valores monetários, entre outros elementos, presentes no texto. A Função "*Clean_BERT*" consiste em uma função para limpeza dos textos que pode ser utilizada em conjunto com o modelo de linguagem BERT (*Bidirecional Encoder Representations from Transformers*, "Representações de Codificadores Bidirecionais de Transformadores").

A seguir são apresentadas comparações do método proposto, utilizando as combinações com desempenho que alcançaram melhores resultados e os métodos de limpeza utilizados na biblioteca do LegalNLP, utilizando como classificador o KNN e apresentando as métricas de acurácia e precisão.

Tabela 19. Comparação com Biblioteca LegalNLP.

Método	**Acurácia Média % (DP)**	**Acurácia Max (%)**	**Precisão Média % (DP)**	**Precisão Max (%)**
LegalNLP (Função Clean)	73.304 (0.03937)	79.368	75.611 (0.05057)	85.012
LegalNLP (Função CleanBERT)	**73.743** (0.04310)	81.528	76.378 (0.05892)	84.962
Método Proposto (T7, T8 e T9)	73.563 (0.04532)	**85.518**	**77.009** (0.05871)	88.858
Método Proposto (T1, T2, T4, T8, T9)	73.671 (0.04112)	83.888	76.691 (0.05812)	**91.149**

Fonte: Autor (2023)

A partir da Tabela 19, é possível perceber que as métricas apresentadas pelo Método Proposto em duas versões de combinações, a primeira com as tarefas T7, T8 e T9, e a segunda com as tarefas T1, T2, T4, T8, T9, conseguem atingir bons resultados em comparação aos métodos do LegalNLP. Apenas para a Acurácia Média o LegalNLP se sobressai em relação aos demais, com **73.743%**, seguido pelo Método Proposto (T7, T8 e T9). Para as demais métricas, o Método Proposto apresenta o maior resultado. Com **85.518%**, o Método Proposto com as tarefas T7, T8 e T9 se destaca para a Acurácia Máxima, assim como se distingue das demais para a Precisão Média, com **77.009%**. Por fim, para a métrica de Precisão Máxima, o Método Proposto (T1, T2, T4, T8, T9) consegue se destacar eficientemente em relação aos demais métodos, apresentando **91.149%**.

5. CONCLUSÃO

A etapa de Preparação de Dados em mineração de textos, se bem estruturada, consegue agregar diversos ganhos para todo o processo de Descoberta de Conhecimento. Como abordado, os dados jurídicos possuem características próprias que devem ser corretamente tratadas para serem inseridas no processo e gerar informações relevantes.

O presente trabalho teve como objetivos realizar análise e avaliação das principais técnicas empregadas no pré-processamento para bases de texto jurídicas, apresentando os resultados dos testes executados. No total, nove tarefas foram testadas individualmente e em conjunto para tratar uma base de dados de petições iniciais.

A combinação entre Remoção de Números, Remoção de Links, Transformação em minúsculas, Stemização, Lematização e Tokenização se apresentou com maior valor de acurácia para seis tarefas, com **81,577%**. Para cinco tarefas, com **83,206%**, a combinação entre Remoção de Números, Remoção de *Links*, Remoção de *Stopwords*, Lematização e Tokenização foi a que conseguiu resultado mais elevado. É importante ressaltar que a combinação que conseguiu alcançar maior valor, **85,518%**, foi uma combinação de três: Stemização, Lematização e Tokenização.

Além disso, foram apontadas recomendações para os erros encontrados no módulo de pré-processamento da biblioteca ia_utils da plataforma SINAPSES/CNJ, em seu Normalizador de Textos (APÊNDICE I). Assim foram apresentadas as tarefas de Generalização de Termos e Extenso Numeral da biblioteca, com a descrição de sua funcionalidade, os erros e possíveis soluções para os problemas.

Assim, as principais contribuições do estudo se refletem na elaboração de fluxos apresentados de pré-processamento que fornecem ganhos significativos para tarefas de Mineração de Textos voltadas para a Área Jurídica, a partir da análise e avaliação das principais técnicas difundidas na literatura, além de demonstrar como a combinação adequada de técnicas podem gerar benefícios para um projeto de Aprendizagem de Máquinas, enquanto uma abordagem falha não gera ganhos relevantes ou até causar prejuízos ao projeto. Outra contribuição do estudo é compreendida pelas recomendações de correção sugeridas para a biblioteca ia_utils da plataforma SINAPSES/CNJ.

Como limitações, é possível perceber que o estudo presente foi baseado e aplicado para um tipo de documento jurídico, Petições Iniciais do Tribunal de Justiça do Estado do Maranhão. Assim, por conta da estrutura textual jurídica brasileira presente nos textos da Base de Dados utilizados, espera-se que seja possível aplicar o método proposto em outros âmbitos e áreas jurídicas, porém

não é possível garantir que o desempenho será efetivamente equivalente para outros contextos e regiões. Outra limitação do estudo se apresenta por não realizar comparações com outros métodos propostos presentes na literatura, dos quais apenas foi citado o LegalNLP.

Nesse caso, como trabalhos futuros, vislumbra-se: 1) Aplicar do Método Proposto em documentos jurídicos de tipos diferentes (não apenas Petições Iniciais); 2) Aplicar o Método Proposto em documentos de outros órgãos da esfera judiciária brasileira (alcançando dialetos fora do Estado do Maranhão); 3) Comparação do Método Proposto com outras bibliotecas, não restringindo-se ao LegalNLP).

REFERÊNCIAS

ALMEIDA, P. A. C. Avaliação de algoritmos de aprendizagem de máquinas na detecção de mutações somáticas. 2019. xv, 51 f., il. Trabalho de Conclusão de Curso (Bacharelado em Engenharia da Computação)—Universidade de Brasília, Brasília.

ANDRADE, P. H. M. A. Aplicação de técnicas de mineração de textos para classificação de documentos: um estudo da automatização da triagem de denúncias na CGU. 2015. xi, 54 f., il. Dissertação (Mestrado Profissional em Computação Aplicada)—Universidade de Brasília, Brasília.

CASTRO JÚNIOR, A. P.; CALIXTO, W. P.; CASTRO, C. H. A. Aplicação da Inteligência Artificial na identificação de conexões pelo fato e tese jurídica nas petições iniciais e integração com o Sistema de Processo Eletrônico. Revista CNJ, Brasília, v. 4, n. 1, p. 9-18, jan./jun. 2020.

CASTRO, P. V. Q. Aprendizagem profunda para reconhecimento de entidades nomeadas em domínio jurídico. 2019. 125 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Goiás, Goiânia, 2019.

CHANDRASEKAR, P; QIAN, K. The impact of data preprocessing on the performance of a naive bayes classifier. In: 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC). IEEE, 2016. p. 618-619.

CIRQUEIRA, D.; JACOB JUNIOR, A. F. L.; LOBATO, F. M. F.; DE SANTANA, A. L.; PINHEIRO, M. Performance Evaluation of Sentiment Analysis Methods for Brazilian Portuguese. Lecture Notes in Business Information Processing. 1ed.: Springer International Publishing, 2017, v. 263, p. 245-251.

CIRQUEIRA, D.; PINHEIRO, M; JACOB JUNIOR, A. F. L.; LOBATO, F. M. F.; DE SANTANA, A. L. A Literature Review in Preprocessing for Sentiment Analysis for Brazilian Portuguese Social Media. IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, Chile, 2018, pp. 746-749.

Coelho, Vinícius & Javidi da Costa, Joao Paulo & Silva, Daniel & de Sousa Junior, Rafael & Mendonça, Fábio Lucio & Silva, Daniel. (2016). MINERAÇÃO DE DADOS EDUCACIONAIS NO ENSINO A DISTÂNCIA GOVERNAMENTAL. CIAWI 2016.

CONSELHO NACIONAL DE JUSTIÇA - CNJ. A Justiça em Números - Relatório Analítico 2022. Brasília, 2022.

D'CASTRO, R. J. Pré-processamento para mineração de processos: técnicas para simplificação automática de logs de eventos. 2020. Tese (Doutorado em Ciências da Computação) - Universidade Federal de Pernambuco, Recife, 2020.

FARACO, F. M. Modelo de conhecimento baseado em tópicos de acórdãos para suporte à análise de petições iniciais. 2020.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From Data Mining to Knowledge Discovery in Databases. AI Magazine, [S. l.], v. 17, n. 3, p. 37, 1996. DOI: 10.1609/aimag.v17i3.1230. Disponível em: https://ojs.aaai.org/index.php/aimagazine/article/view/1230.

FERREIRA, M. H. P. Classificação de peças processuais jurídicas: Inteligência Artificial no Direito. 2018.

FREITAS JUNIOR, A. A. Um estudo de algoritmos de aprendizagem de máquinas para o Smart Defender. 2020. 54 f. Trabalho de Conclusão de Curso Graduação em Engenharia de Computação) - Centro de Tecnologia, Universidade Federal do Rio Grande do Norte, Natal, 2020.

GARCÍA, S; LUENGO, J; HERRERA, F. Data preprocessing in data mining. Cham, Switzerland: Springer International Publishing, 2015.

GONÇALVES, M. C.; SOUZA, M. W. A.. Tópicos em aprendizagem estatística de máquinas com aplicações em finanças. Niterói, 2020. 61 f.

GRANCHAROVA, M.; JANGEFALK, M. Comparative Study of the Combined Performance of Learning Algorithms and Preprocessing Techniques for Text Classification. 2018. , p. 29.

GUSMÃO, C.; FIGUEIREDO, K.; BRITO, W. A. T. Técnicas de Processamento de Linguagem Natural em Denúncias Criminais: Automatização e Classificação de Texto em Português Coloquial. In: Anais do XLVIII Seminário Integrado de Software e Hardware. SBC, 2021. p. 172-182.

HAGEN, L. Content analysis of e-petitions with topic modeling: How to train and evaluate LDA models? Information Processing and Management, v. 54, n. 6, p. 1292–1307, 2018. DOI: 10.1016/j.ipm.2018.05.006. Disponível em: https: //www.scopus.com/inward/record.uri?eid=2-s2.0-85047760326&doi=0.1016%2fj.ipm.2018.05.006&partnerID=40&md5=e6c91147e78a3e72bb20678c67404ae5.

IŞIK, M; DAĞ, H. The impact of text preprocessing on the prediction of review ratings. Turkish Journal of Electrical Engineering and Computer Sciences, v. 28, n. 3, p. 1405-1421, 2020.

JUNIOR, R.; DE MEDEIROS MELO, W.; DE ARAUJO FAGUNDES, R.; ANDRADE MACIEL, A. Extração de Informação e Mineração de Dados no Diário Oficial de Pernambuco. Revista de Engenharia e Pesquisa Aplicada, v. 3, n. 3, 30 ago. 2018.

KADHIM, A. I. An evaluation of preprocessing techniques for text classification. International Journal of Computer Science and Information Security (IJCSIS), v. 16, n. 6, p. 22-32, 2018.

LJUNG, Kevin. Clean Code in Practice: Developers perception of clean code. 2021.

MACHADO, G. M. Um estudo da representação de documentos jurídicos em espaços métricos. 2019. 74 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Sergipe, São Cristóvão, SE, 2019.

MACIEL, T.; SEUS, V.; MACHADO, K.; BORGES, E. Mineração de dados em triagem de risco de saúde. Revista Brasileira de Computação Aplicada, v. 7, n. 2, p. 26-40, 15 maio 2015.

MADEIRA, R. O. C. Aplicação de técnicas de mineração de texto na detecção de discrepâncias em documentos fiscais. 2015. Tese de Doutorado.

MASTELLA, J. O. Uma metodologia usando ambientes paralelos para otimização da classificação de textos aplicada a documentos jurídicos. 2020. Dissertação de Mestrado. Pontifícia Universidade Católica do Rio Grande do Sul.

MATSUI, B. M. A; GOYA, D. H. Applying DevOps to Machine Learning Processes: A Systematic Mapping. In: ENCONTRO NACIONAL DE INTELIGÊNCIA ARTIFICIAL E COMPUTACIONAL (ENIAC), 18. , 2021, Evento Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 559-570. DOI: https://doi.org/10.5753/eniac.2021.18284.

PEREIRA, J. C. M.; RODRIGUES, M. V. J. A Plataforma Sinapses e a Continuidade dos Modelos de IA no Judiciário. 2022.

POLO, F. M., MENDONÇA, G. C. F., PARREIRA, K. C. J., GIANVECHIO, L., CORDEIRO, P., FERREIRA, J. B., VICENTE, R. (2021). LegalNLP - Natural Language Processing methods for the Brazilian Legal Language. arXiv preprint arXiv:2110.15709.

RAMINELLI, D. G. T. L.; SANTOS, B. S. Aplicação de Técnicas de Mineração de Dados e Aprendizagem de Máquina no Mercado de Ações: Uma Revisão Sistemática. In: Congresso Brasileiro de Engenharia de Produção. 2019.

RIBEIRO, E. R. Impacto de técnicas de pré-processamento de texto na detecção de intenção e extração de parâmetros em sistemas de diálogo orientados a tarefa. 2020. 64 f. Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas, Manaus, 2020.

SALEEM, A; ASIF, K. H.; ALI, A.; AWAN, S. M.; ALGHAMDI, M. A.. Pre-processing methods of data mining. In: 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing. IEEE, 2014. p. 451-456.

SILVA, J. A.; NOGUEIRA J. R.; V., OLIVEIRA, H.; BARBOSA, A.; VIEIRA, T.; OLIVEIRA, K. Avaliação de abordagens para classificação automática de documentos jurídicos: um estudo comparativo aplicado a petições do Tribunal de Justiça do Estado de Alagoas. Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 8, n. 1, 2021.

SOUSA, R. N. MINERJUS: solução de apoio à classificação processual com uso de Inteligência Artificial. 2019. 59f. Dissertação (Mestrado em Modelagem Computacional e

Sistemas) – Universidade Federal do Tocantins, Programa de Pós-graduação em Modelagem Computacional e Sistemas, Palmas, 2019.

WIRTH, R.; HIPP, J. Crisp-dm: towards a standard process model for data mining. 2020.

APÊNDICE

APÊNDICE I - Relatório de Bugs Encontrados na Biblioteca de Pré-Processamento do Sinapses/CNJ

UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS
MESTRADO EM ENGENHARIA DE COMPUTAÇÃO E SISTEMAS
TRIBUNAL DE JUSTIÇA DO ESTADO DO MARANHÃO

RELATÓRIO DE BUGS ENCONTRADOS NA BIBLIOTECA IA_UTILS DISPONÍVEL PELO SINAPSES/CNJ

São Luís, 03 de setembro de 2023

FICHA TÉCNICA DO RELATÓRIO

EQUIPE EXECUTORA

Universidade Estadual do Maranhão - UEMA

Professores-Pesquisadores

MSc. Antonio Fernando Lavareda Jacob Junior (Coordenador)

Dr. Fábio Manoel França Lobato

Dr. Ewaldo Eder Carvalho Santana

Alunos de Mestrado

Diógenes Ademir Domingos

Fabrício Almeida do Carmo

Marcos Vinicius Januário da Silva

Matheus Serrão Marinato

Alunos de Graduação

Aline Mariana Barros Silva

Joerllen Melonio Cutrim

Jonas Carvalho De Sousa Neto

Marcio Vinicius da Silva dos Santos

Thyago Machado Rodrigues

Victor Emanuel Santos Moura

Tribunal de Justiça do Estado do Maranhão - TJMA

Juízes

MMº. Juiz de Direito Ferdinando Marco Gomes Serejo Sousa

MMº. Juiz de Direito Raniel Barbosa Nunes

Coordenadoria de Sistemas de Informação

Francisco de Araújo Costa

Januário da Silva, Marcos Vinicius; Jacob Jr, Antonio Fernando Lavareda; Lobato, Fábio Manoel França; Santana, Ewaldo Eder Carvalho.

Relatório de Bugs Encontrados na Biblioteca ia_utils disponível pelo Sinapses/CNJ./ Marcos Vinicius Januário da Silva, Antonio Fernando Lavareda Jacob Jr, Fábio Manoel França Lobato, Ewaldo Eder Carvalho Santana. – São Luís, 2023.

12f.

Orientador: Prof. Msc. Antonio Fernando Lavareda Jacob Junior

Relatório Técnico - Universidade Estadual do Maranhão - UEMA / Tribunal de Justiça do Estado do Maranhão.

Resumo

Neste relatório técnico é apresentada uma descrição dos problemas encontrados na biblioteca ia_utils disponível pelo Sinapses/CNJ que é focada no pré-processamento de texto. As tarefas que foram encontrados erros de execução foram a tarefa de Generalização de Termos e a tarefa de Extração de Numeral na sua forma por extenso. Para analisar os erros foram realizados teste nas tarefas, que demonstraram a falha. No entanto, são mostradas formas de contorna esses problemas aplicando métodos e tarefas adicionais.

Palavras-chave: Inteligência Artificial, Bugs, Processamento de Textos.

SUMÁRIO

1 INTRODUÇÃO

O pré-processamento de texto é frequentemente o primeiro passo no pipeline de um sistema de Processamento de Linguagem Natural (PLN), com impacto potencial em seu desempenho final (CAMACHO-COLLADOS e PILEHVAR, 2017). O pré-processamento de texto é o processo pelo qual limpamos os dados brutos de texto, removendo o ruído, como pontuações, emojis e palavras comuns, para prepará-lo para o treinamento de algum modelo. É muito importante remover dados ou partes não úteis do nosso texto[1].

Os dados que são obtidos para o uso, geralmente são não estruturados. Contém texto e símbolos incomuns que precisam ser limpos para que um modelo de aprendizado de máquina possa compreendê-lo. A limpeza e o pré-processamento de dados são tão importantes quanto a construção de qualquer modelo sofisticado de aprendizado de máquina[2].

Portanto, é seguro dizer que o pré-processamento de texto é uma etapa essencial no PLN que envolve a limpeza e transformação de dados de texto não estruturados para prepará-los para análise. Em vista disso o objetivo deste documento é relatar erros encontrados ao se utilizar funções disponíveis no Sinapses/CNJ focadas para pré-processamento de texto, apresentando também possíveis soluções de implementações.

O Sinapses/CNJ disponibiliza para importação e utilização a biblioteca ia_utils, possuindo um Normalizador de Texto, que abrange algumas tarefas de para realizar ajustes em base de dados textuais. As tarefas que foram encontrados erros de execução foram a tarefa de Generalização de Termos e a tarefa de Extração de Numeral na sua forma por extenso.

[1] https://www.pluralsight.com/guides/importance-of-text-pre-processing
[2] https://www.scaler.com/topics/nlp/text-processing-in-nlp/

2 BUGS ENCONTRATOS

2.1 Bug 1 - Tarefa de Generalização (ia_utils)

2.1.1 Descrição do Bug

A Tarefa de Generalização de Termos deve gerar como resultado uma normalização das palavras dentro de uma sentença. A tarefa está presente na biblioteca ia_utils como ExtratorGeneralizacao. Porém, esta tarefa não está apresentando os resultados esperados de generalização. Abaixo são demonstrados os erros encontrados e uma possível solução para o problema.

2.1.2 Análise do Bug

A figura 1 demonstra o resultado utilizando a Generalização de Termos da biblioteca ia_utils do Sinapses/CNJ. Para realizar os testes, foi utilizado a frase: “Pesquisa revela que 10% da população possui 75% da riqueza mundial.”.

Figura 1 - Generalização de Termos do Normalizador da ia_utils do Sinapses/CNJ

```
from ia_utils.extrator_generalizacao.extrator_generalizacao import ExtratorGeneralizacao
extrator_generalizacao = ExtratorGeneralizacao()
subtexto_ia_utils_generalizacao = extrator_generalizacao.substituir_no_texto(texto)
subtexto_ia_utils_generalizacao
```

```
'Pesquisa revela que 10% da população possui 75% da riqueza mundial.'
```

Fonte: Composição Própria (2023).

2.2 Bug 2 - Tarefa de Extenso Numeral (ia_utils)

2.2.1 Descrição do Bug

A Tarefa de Extenso Numeral deve gerar como resultado a conversão de números para a sua forma escrita por extenso dentro de uma sentença. A tarefa está presente na biblioteca ia_utils como extrator_extenso_numeral. Porém, esta tarefa não está apresentando os resultados esperados de transformação para extenso. Abaixo são demonstrados os erros encontrados e uma possível solução para o problema.

2.2.2 Análise do Bug

A figura 6 demonstra o resultado utilizando a tarefa de Extenso Numeral da biblioteca ia_utils do Sinapses/CNJ. Para realizar os testes, foi utilizado um texto de exemplo contendo números: “4568378403”.

Figura 6 - Tarefa de Extenso Numeral do Normalizador da ia_utils do Sinapses/CNJ

```
from ia_utils.extrator_extenso_numeral import extrator_extenso_numeral
subtexto_ia_utils_extenso_numeral = extrator_extenso_numeral.substituir_no_texto(texto)
subtexto_ia_utils_extenso_numeral
```

```
'4568378403'
```

Fonte: Composição Própria (2023).

3 RESULTADOS E DISCUSSÃO

3.1 Solução do Bug da Tarefa de Generalização (ia_utils)

A seguir, a figura 2 demonstra o resultado utilizando a tarefa de Lematização como tarefa de Pré-processamento.

Figura 2 - Utilizando Lematização como tarefa de Pré-processamento

```
subtexto_uema_utils_lema = lematizacao(texto)
subtexto_uema_utils_lema
```

```
'pesquisa revelar que 10 % de o população possuir 75 % de o riqueza mundial . '
```

Fonte: Composição Própria (2023).

A seguir, a figura 3 demonstra o resultado utilizando a tarefa de Stemização como tarefa de Pré-processamento.

Figura 3 - Utilizando Stemização como tarefa de Pré-processamento

```
subtexto_uema_utils_stem = stemizacao(texto)
subtexto_uema_utils_stem
```

```
'pesquis revel que 10 % da popul possu 75 % da riqu mund . '
```

Fonte: Composição Própria (2023).

A tabela 1 demonstra comparação entre os resultados utilizando as três abordagens. Como observado, a única que não alterou o texto original em nenhuma forma foi a Generalização de Termos do Normalizador da biblioteca ia_utils do Sinapses/CNJ.

Tabela 1. Comparação de Resultados - Tarefa de Generalização, Lematização e Stemização

Texto Original	“Pesquisa revela que 10% da população possui 75% da riqueza mundial.”
Generalização de Termos do Normalizador da ia_utils do Sinapses/CNJ	“Pesquisa revela que 10% da população possui 75% da riqueza mundial.”

Lematização da biblioteca de Pré-processamento comparativa	“pesquisa revelar que 10 % de o população possuir 75 % de o riqueza mundial .”
Stemização da biblioteca de Pré-processamento comparativa	“pesquis revel que 10 % da popul possu 75 % da riqu mund .”

Fonte: Composição Própria (2023).

3.1.1 Recomendações

Para a tarefa de Generalização de Termos da plataforma Sinapses/CNJ alcançar seu objetivo é necessário alterar sua construção. Dessa forma, recomenda-se utilizar uma das tarefas demonstradas anteriormente, Lematização e Stemização, de acordo com a avaliação de qual se encaixa melhor para o contexto. Ou deixar disponíveis as duas abordagens para serem utilizadas, que é o mais recomendado. Abaixo estão códigos de exemplos para sua construção utilizando bibliotecas do Python.

Para a utilização da tarefa de Lematização é recomendado utilizar a estrutura a seguir apresentada na Figura 4:

Figura 4 - Estrutura da tarefa de Lematização

```
import spacy
nlp = spacy.load("pt_core_news_sm")
def lematizacao(texto):
    texto = nlp(texto)
    texto_lematizado = []
    for token in texto:
        texto_lematizado.append(token.lemma_)
        texto_lematizado.append(" ")
    texto = "".join(texto_lematizado)
    return texto
```

Fonte: Composição Própria (2023).

Para a utilização da tarefa de Stemização é recomendado utilizar a estrutura a seguir apresentada na Figura 5:

Figura 5 - Estrutura da tarefa de Stemização

```
from nltk.tokenize import word_tokenize
from nltk.stem import RSLPStemmer
def stemizacao(texto):
    stemizador = RSLPStemmer()
    tokens = word_tokenize(texto)
    texto_stemizado = []
    for t in tokens:
        texto_stemizado.append(stemizador.stem(t))
        texto_stemizado.append(" ")
    texto = "".join(texto_stemizado)
    return texto
```

Fonte: Composição Própria (2023).

3.1.2 Solução do Bug da Tarefa de Extenso Numeral (ia_utils)

A seguir, a figura 7 demonstra o resultado utilizando a tarefa de Remoção de Números como tarefa de Pré-processamento.

Figura 7 - Utilizando Remoção de Números como tarefa de Pré-processamento

```
subtexto_remover_numeros = remover_numeros(texto)
subtexto_remover_numeros
```

```
''
```

Fonte: Composição Própria (2023).

A seguir, a figura 8 demonstra o resultado utilizando a tarefa de Transformação para Forma Por Extenso como tarefa de Pré-processamento.

Figura 8 - Utilizando Transformação para Forma Por Extenso como Pré-processamento

```
subtexto_extenso_numeral = por_extenso_numeros(texto)
subtexto_extenso_numeral
```

```
'quatro mil quinhentos e sessenta e oito milhões trezentos e setenta e oito mil quatrocentos e três'
```

Fonte: Composição Própria (2023).

A tabela 2 demonstra comparação entre os resultados utilizando as três abordagens. Como observado, a única que não alterou o texto original em nenhuma forma foi a tarefa de Extenso Numeral do Normalizador da biblioteca ia_utils do Sinapses/CNJ.

Tabela 2. Comparação de Resultados - Tarefa de Extenso Numeral, Remoção de Números e Transformação para Forma Por Extenso

Texto Original	“4568378403”
Extenso Numeral do Normalizador da ia_utils do Sinapses/CNJ	“4568378403”
Remoção de Números da biblioteca de Pré-processamento comparativa	“”
Transformação para Forma Por Extenso da biblioteca de Pré-processamento comparativa	“quatro mil quinhentos e sessenta e oito milhões trezentos e setenta e oito mil quatrocentos e três”

Fonte: Composição Própria (2023).

3.2.1 Recomendações

Para a tarefa de Extenso Numeral da plataforma Sinapses/CNJ alcançar seu objetivo é necessário alterar sua construção. Dessa forma, recomenda-se utilizar uma das tarefas demonstradas anteriormente, Remoção de Números e Transformação para Forma Por Extenso, de acordo com a avaliação de qual se encaixa melhor para o contexto.

Outra opção é disponibilizar as duas abordagens para serem utilizadas, que é o mais recomendado. Abaixo estão códigos de exemplos para sua construção utilizando bibliotecas do Python.

Para a utilização da tarefa de Remoção de Números é recomendado utilizar a estrutura a seguir apresentada na Figura 9.

Figura 9 - Estrutura da tarefa de Remoção de Números

```
import re
def remover_numeros(texto):
    texto = str(texto)
    texto = re.sub(r'[0-9]+', r'', texto)
    return texto
```

Fonte: Composição Própria (2023).

Para a utilização da tarefa de Transformação para Forma Por Extenso é recomendado utilizar a estrutura a seguir apresentada na Figura 10:

Figura 10 - Estrutura da tarefa de Transformação para Forma Por Extenso

```
def por_extenso_numeros(texto):
    texto = num2words(texto, lang='pt-br')
    return texto
```

Fonte: Composição Própria (2023).

4 CONCLUSÕES

Como pode ser analisado, as tarefas de Generalização de Termos (ExtratorGeneralizacao) e de Extenso Numeral (extrator_extenso_numeral) presentes no Normalizador de Texto da biblioteca ia_utils do Sinapses/CNJ não executam sua função como esperado, não resultando em nenhum efeito no texto original empregado. É recomendado seguir as sugestões de bibliotecas com funções iguais ou semelhantes, descritas nos itens anteriores.

REFERÊNCIAS

CAMACHO-COLLADOS, J.; PILEHVAR, M. T. On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis, 2017.