O Uso da Mineração de Processos na Análise do Tempo das Movimentações Processuais

The Use of Process Mining in Time Analysis of Process Movements

Jaqueline K. L. da Cruz[1]
orcid.org/0009-0005-2046-0498

Luiz F. V. Verçosa[1]
orcid.org/0000-0003-2095-9000

Vinícius Ferreira Silva [3]
orcid.org/0000-0001-9889-2331

Carmelo J. A. Bastos-Filho[1]
orcid.org/0000-0002-0924-5341

Byron L. Dantas Bezerra[1]
orcid.org/0000-0002-9327-9734

[1]Escola Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, Brasil. E-mail: jklc@ecomp.poli.br

DOI: 10.25286/repa.v9i1.2785

Como citar este artigo pela NBR 6023/2018: Jaqueline K. L. da Cruz; Luiz F. V. Verçosa; Vinícius Ferreira Silva; Carmelo Bastos-Filho; Byron L. Dantas Bezerra. Sistema de O Uso da Mineração de Processos na Análise do Tempo das Movimentações Processuais. Revista de Engenharia e Pesquisa Aplicada, v.9, n. 1, p. 97-104, 2024. DOI: 10.25286/repa.v9i1.2785

RESUMO

O tempo exigido para resolução de processos judiciais impacta a economia, a confiança dos investidores em um país e a vida da população. Trabalhos da literatura fornecem soluções nesse sentido, porém não consideram as movimentações processuais realizadas no processo durante seu trâmite. Este trabalho utiliza modelos de aprendizagem de máquina treinados com características pautadas na sequência de movimentações processuais e características do tribunal responsável pelo processo. Foram utilizados processos eleitorais de diferentes tribunais regionais, TREs, de estados do nordeste brasileiro. Os modelos foram capazes de prever o tempo total processual com alta acurácia em que o Light Gradient Boosting Machine (LGBM) obteve R^2 médio de 0.9. Os resultados sugerem a eficiência da abordagem e a exploração de técnicas de mineração de processos como diferencial para a tarefa.

PALAVRAS-CHAVE: Jurimetria; Aprendizagem de Máquina; Ciência dos Dados.

ABSTRACT

The time required to finish legal proceedings impacts the economy, investor confidence in a country and the lives of the population. Literature works provide solutions in this sense, but do not consider the procedural movements carried out in the process during its execution. This work uses machine learning models trained with characteristics based on the sequence of procedural movements and characteristics of the court responsible for the process. There were used lawsuits from multiple northeast regional labor courts in Brazil (TRE). The models were able to predict the total procedural time with high accuracy in which the Light Gradient Boosting Machine (LGBM) obtained an average R^2 of 0.9. The results suggest the efficiency of the approach and the success of process mining techniques for the task

KEY-WORDS: Jurimetrics; Machine Learning; Data Science.

1 INTRODUÇÃO

A qualidade de um sistema judicial pode ser medida pela equidade, eficiência na alocação de recursos e produtividade. Ineficiências impactam a economia, além da percepção de confiança de investidores estrangeiros **[1]**.

Exemplo de baixa performance pode refletir-se, por exemplo, em processos judiciais de longa duração. O Conselho Europeu no Artigo 6 da Convenção Europeia de Direitos Humanos **[2]** determina que os tribunais finalizem os casos judiciais em tempo razoável, embora o termo "razoável" seja vago e objeto de discussão. No Brasil, a problemática é semelhante já que são considerados princípios de duração razoável e celeridade **[3]**. Além disso, relata-se a percepção de morosidade com relação aos processos judiciais **[4]** por parte da população. Isso pode estar relacionado a alguns aspectos como o número menor de magistrados *per capita* quando comparado com outros países e o alto número de ritos burocráticos **[4]**.

Abordagens que utilizam aprendizagem de máquina vêm sendo utilizadas em tarefas correlatas para predição do tempo processual **[5]** e resultados processuais **[6]**. Entretanto, o conjunto de *features* utilizadas pode variar substancialmente revelando diferentes aspectos dos processos investigados. Neste trabalho, é investigada a habilidade de diferentes modelos de aprendizagem de máquina para prever o tempo total processual. São utilizados 8.186 processos judiciais do trabalho, TREs, obtidos de vários estados do nordeste brasileiro. Além disso, são identificados os conjuntos de *features* mais promissores para o modelo.

A continuação deste trabalho encontra-se estruturada da seguinte maneira. A Seção 2 exibe trabalhos relacionados e revela as novidades da presente proposta. A Seção 3 descreve a metodologia utilizada, pormenorizando detalhes referentes à base de dados, criação de *features*, características dos modelos de aprendizagem de máquina e métricas utilizadas. A Seção 4 apresenta os resultados dos modelos utilizados e a análise de importância das *features* mais relevantes. A Seção 5 conclui o trabalho.

2 TRABALHOS RELACIONADOS

Jurimetria é um termo que se refere à aplicação de métodos estatísticos e matemáticos ao entendimento dos processos e fatos jurídicos **[7]**. Sob esta ótica, alguns trabalhos se dedicam à previsão do resultado de um processo judicial **[6,8]**, utilizando modelos de aprendizagem de máquina. Em **[6]** os autores utilizam técnicas de mineração de textos e dados administrativos para criação de *features* que servirão de entrada para diferentes modelos de aprendizagem de máquina na tarefa de previsão de resultados em processos judiciais civis. Em **[9]** os autores realizam uma revisão da literatura a respeito de técnicas de aprendizagem de máquina que exploram essa direção. Técnicas de mineração de processos permitem a previsão de resultado ou tempo restante em processos em andamento **[10,11]**. Em **[11]** os autores propõem um método para direcionamento de caminho de um processo em execução de forma que metas de negócio definidas pelo usuário possam ser alcançadas. Em contrapartida, o presente trabalho visa a previsão de tempo de um processo já finalizado, i.e., análise *post-mortem*. Outros trabalhos também focam na perspectiva de mineração de processos para avaliar causas de ineficiência em tribunais brasileiros **[12,13]**. Por exemplo, em **[12]** os autores identificam os principais gargalos processuais no Tribunal de Justiça de São Paulo - Brasil. Os trabalhos **[14]** e **[5]** dedicam-se à previsão do tempo processual de processos judiciais utilizando técnicas de aprendizagem de máquina e se assemelham ao presente trabalho. Um diferencial deste trabalho está na natureza das *features* exploradas. Utiliza-se aqui as movimentações processuais executadas nos processos durante seu trâmite, i.e., considera-se todo o curso processual. A premissa é que não só a natureza das movimentações envolvidas, mas também a sua frequência e ordem podem impactar no tempo processual total.

3 METODOLOGIA

O tempo total processual foi considerado como a diferença em dias entre o tempo associado à última movimentação processual e o tempo referente à primeira movimentação processual encontrados para o processo.

Os códigos foram desenvolvidos na linguagem de programação Python com a utilização da biblioteca sklearn para a aplicação de modelos de aprendizagem de máquina. Todos os códigos estão disponíveis no github aberto dos autores[1].

3.1 BASE DE DADOS

A base de dados foi criada a partir dos dados disponibilizados pelo Conselho Nacional de Justiça (CNJ) durante uma competição *hackathon* realizada em 2020[2]. A base contém exemplos de processos judiciais em andamento ou finalizados de diversas esferas judiciais, e.g., *Tribunais do Trabalho, Tribunais Eleitorais, Supremo Tribunal de Justiça*, dentre outros.

Os dados disponibilizados pelo CNJ referem-se ao cabeçalho e movimentações processuais. O cabeçalho identifica a classe e os assuntos processuais. A classe processual se refere ao procedimento adotado na esfera judicial para o processo, e.g., *Filiação Partidária, Ação Penal Eleitoral*. Os assuntos processuais referem-se a temas e matérias discutidas durante o desenvolvimento do processo, e.g., *Cancelamento de Filiação Partidária*, *Impugnação da Inscrição Eleitoral*. Por sua vez, as movimentações processuais identificam etapas relevantes percorridas pelo processo durante o seu trâmite, e.g., *Julgamento*, *Trânsito em Julgado*, *Conclusão*. Além desses dados, foram obtidos documentos públicos disponibilizados pelo CNJ que contêm características dos tribunais, i.e., Unidades Julgadoras (UJs). Esses dados são referentes à classificação da UJ e ao seu nível de congestionamento.

Foram utilizados processos judiciais referentes a tribunais eleitorais dos estados de Pernambuco (17%), Ceará (21%), Maranhão (1%), Paraíba (5%), Rio Grande do Norte (12%), Piauí (15%), Alagoas (9%) e Sergipe (20%). No total esses processos somaram 8.186 casos judiciais com movimentações compreendidas entre os anos 2000 e 2017. Processos com movimentações ocorridas entre 2018 e 2020 foram removidos (o que foi o caso de todos os processos da Bahia). A região nordeste foi escolhida por sua similaridade cultural e econômica entre seus estados integrantes.

A Figura 1 exibe o tempo total para os processos judiciais analisados em número de anos. Esses tempos variam de alguns meses até em torno de nove anos. Percebe-se uma certa quantidade de processos com tempo inferior a um ano, i.e., 15%, entretanto, a maioria concentra-se no intervalo entre 3 a 8 anos, i.e., 73%.

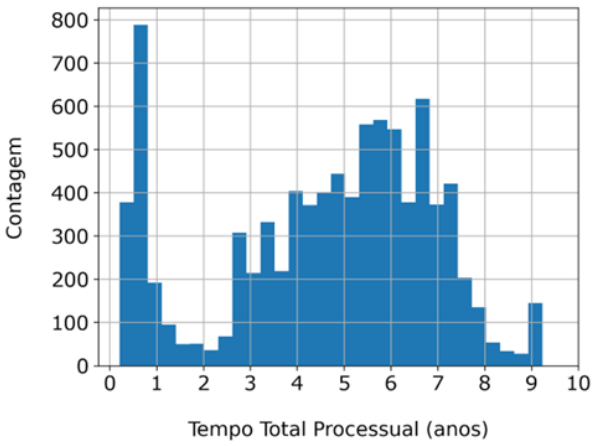


Figura 1 – Tempo processual total para os processos judiciais analisados em anos.

Fonte: Os autores.

3.2 CRIAÇÃO DE FEATURES

Após tratamento dos dados, foram criadas 52 *features* e um alvo. As *features* foram divididas em dois grupos exibidos na Tabela 1. O alvo é o tempo total processual medido em número de dias.

Tabela 1 - Descrição das *features* presentes na base de dados.

Categoria	Tipo	Quantidade
UJ	Congestionamento	15
	Região	2
	Especificação	2
Processo	Movimentações	29
	Classe	1
	Assunto	2
	Digital	1

Fonte: Os autores.

No total, constam 52 *features* agrupadas nas categorias UJ e Processo. O agrupamento foi realizado com o objetivo de condensar as informações referentes às características das *features* utilizadas na análise. A categoria Unidade Judiciária (UJ) refere-se a aspectos da UJ responsável pelo processo. Ela foi distribuída em três *Tipos* que são: *Congestionamento*, *Região* e *Especificação*. As *features* de congestionamento totalizam 15 e especificam o nível de acúmulo de processos na UJ em diferentes anos e para diferentes tipos de processos, e.g., *Criminais* e *Não-Criminais*. Ela se divide principalmente em *features* que contabilizam o número de processos *Pendentes* em diferentes anos, a *Taxa de Congestionamento*

DOI: 10.25286/repa.v9i1.2785

identificada e os *Recursos Internos* disponíveis. As duas *features* de região distinguem a unidade federativa da UJ e apresentam um código único identificador. O motivo de manter esse identificador único é pela possibilidade de UJs semelhantes possuírem códigos também semelhantes. As duas *features* de *Especificação* especificam se a UJ é de primeira ou segunda instância e se é classificada como uma *Zona Eleitoral*, *Gabinete*, dentre outros.

As *features* da categoria *Processo* expressam diretamente características do processo judicial. O tipo *Movimentações* contém *features* obtidas a partir das movimentações processuais presentes no processo. Para a criação das *features* do tipo *Movimentações* foram utilizados a técnica de *n-gram* e hierarquia de movimentações presentes na base de dados.

A técnica de *n-gram* é amplamente utilizada na literatura em tarefas de mineração de texto e em mineração de processos **[15]**. Ela consiste em encontrar sequências de caracteres (ou palavras) dentro de um texto. A Tabela 2 ilustra a aplicação de *n-gram* para criação de *features* a partir da sequência de caracteres da coluna *Traço*.

Tabela 2 - Exemplo de *n-gram* com *n = 1* para sequências de movimentações processuais codificadas por letras**.**

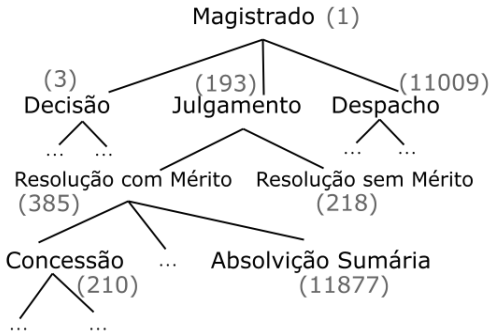
ID Processo	Traço	a	b	c	d
1	⟨a,a,b,c⟩	2	1	1	0
2	⟨a,b,c,b,c,d⟩	1	2	2	1

Fonte: Os autores.

O valor de *n* refere-se ao comprimento da sequência a ser verificada no traço por meio de uma janela deslizante. Com *n = 1*, conta-se apenas o número de ocorrências do caractere no traço. Por exemplo, para o traço ⟨a,a,b,c⟩ conta-se a ocorrência de dois caractere 'a', um caractere 'b' e um caractere 'c'. O *n-gram* com *n = 1* foi utilizado nos experimentos para criação de *features*. Nesse caso, o caractere se refere a uma movimentação processual no traço. O traço, por sua vez, identifica a sequência ordenada de movimentações processuais para um processo, e.g., ⟨Conclusão, Julgamento, Publicação⟩. A ordenação é feita através do atributo *Tempo* presente para cada movimentação processual executada para o processo judicial. Dessa forma, as *features* de *Movimentação* identificam o número de ocorrências de diferentes movimentações processuais em cada instância de processo judicial.

A hierarquia de movimentações está presente na base de dados em estruturas de árvore. A Figura 2 ilustra essa hierarquia para atividades executadas pelo magistrado.

Figura 2 – Ilustração de hierarquia de movimentações processuais executadas pelo magistrado.



Fonte: Os autores.

A relevância da hierarquia está na possibilidade de mapeamento de movimentações processuais mais específicas, para sua equivalente mais genérica, i.e., "ancestral" na hierarquia. Por exemplo, a movimentação processual *Concessão* pode ser mapeada para *Julgamento*. Esse mapeamento foi utilizado para evitar a "explosão" de *features* de movimentações, já que havia no total 210 movimentações utilizadas. Com o mesmo propósito, as *features* de Classe e Assunto processual foram mapeadas para seu nível mais genérico utilizando as suas respectivas árvores.

3.3 PRÉ-PROCESSAMENTO

Múltiplas técnicas de pré-processamento foram aplicadas com o objetivo de garantir a qualidade dos dados a serem fornecidos aos modelos. Registros contendo atributos nulos ou inválidos foram descartados. Foram considerados inválidos, valores dos atributos classe, assunto ou movimentação processual ausentes nas tabelas de referência do CNJ. Foram removidos processos judiciais com movimentação única, processos judiciais com duração total incomuns (menor que 3% percentile ou superior a 97% percentile), processos judiciais com movimentações incomuns de início ou fim (ocorrência inferior a 1% dos casos) e processos com movimentações em anos superiores a 2017. Esse último filtro foi realizado a fim de evitar processos ainda em andamento já que os processos remanescentes permaneceram três anos sem ocorrência de qualquer movimentação processual (os dados foram coletados no segundo semestre de 2020 pelo CNJ), o que foi assumido como processos finalizados.

3.4 MODELOS

Foram aplicados quatro modelos distintos de aprendizagem de máquina. O primeiro modelo é um modelo ingênuo que sempre utiliza o valor médio do alvo da base de treino como previsão. Tal modelo é útil para mensurar a melhoria de performance de modelos inteligentes. O segundo modelo utilizado foi a regressão linear que é relevante por sua simplicidade. Em seguida, empregou-se o modelo *Light Gradient Boosting Machine* (LGBM) que é um modelo que utiliza *gradient boosting* com árvores de decisão. Esse modelo foi proposto pela Microsoft com o intuito de alcançar resultados competitivos com outras técnicas de *boosting* de maneira mais eficiente e escalável **[16]**. O *boosting* é uma técnica utilizada durante o treinamento dos classificadores, i.e., geralmente *week learners,* do modelo e que consiste em atribuir diferentes pesos às instâncias a serem classificadas. Instâncias classificadas erroneamente têm seus pesos incrementados para aumentar a probabilidade de serem classificadas corretamente por novos classificadores. Uma vantagem do LGBM é que por utilizar árvores de decisão como classificadores/regressores possui explicabilidade dos resultados. É possível, assim, determinar as *features* responsáveis pelos principais ganhos (redução em erro, e.g., índice gini) durante o treinamento. A última técnica utilizada foi o *Support Vector Regressor* (SVR). O SVR corresponde a uma adaptação do *Support Vector Machine* (SVM) para tarefas de regressão **[17]**. O SVM é amplamente conhecido por sua alta capacidade de generalização e eficácia em tarefas de classificação **[17]**. A desvantagem do SVR refere-se à interpretação dos resultados, pois os mecanismos aplicados nessa técnica não permitem que as *features* mais relevantes sejam facilmente identificadas.

A Tabela 3 exibe os resultados finais da configuração de parâmetros para os modelos SVR e LGBM. Os parâmetros não-exibidos foram utilizados em sua versão *default* presente no *sklearn*. A regressão linear, i.e., modelo RL, não possui parâmetros a serem otimizados na biblioteca utilizada, i.e., *sklearn* v 1.0.2. O parâmetro *C* do SVR define a intensidade do termo de regularização, i.e., inversamente proporcional ao *C*. O kernel é parâmetro vital do SVR e permite o mapeamento das *features* de entrada em um espaço de maior dimensionalidade. O parâmetro *gamma* permite refinamento através da definição de coeficientes para o kernel.

Tabela 3 - Configurações paramétricas finais dos modelos**.**

Modelo	Parâmetros otimizados
SVR	C: 1024
	kernel: rbf
	gamma: scale
LGBM	boosting_type: dart
	learning_rate: 0.2
	n_estimators: 600

Fonte: Os autores.

Tratando-se do LGBM, o parâmetro *boosting_type* define o mecanismo de *gradient boosting* a ser utilizado. O parâmetro *learning_rate* refere-se à taxa de aprendizagem e n_estimators especifica o número de *week learners* presentes no modelo.

3.5 MÉTRICAS

Os modelos foram avaliados utilizando-se validação *k-fold* com *k* = 10. Além disso, foram aplicadas três métricas distintas para avaliação dos modelos, apresentadas a seguir.

- Coeficiente de determinação (R^2): Também conhecido por R^2 *score*, determina a proporção de variação na variável dependente que pode ser explicada pelas variáveis independentes. Em outras palavras, essa métrica captura o *fitness*, isto é, a proximidade entre as predições e os valores reais do alvo (um R^2 *score* igual a um indica que todos os valores previstos foram exatamente iguais aos valores reais). Na Equação **(1)** o numerador do termo fracionário corresponde aos resíduos de predição, e o denominador corresponde aos resíduos de predição que utiliza sempre o valor médio da base de treino.

$$\textbf{(1)}\quad R^2 = 1 - \frac{\sum_{j=1}^{n}(y_j - p_j)^2}{\sum_{j=1}^{n}(y_j - \bar{y})^2}$$

- Erro médio absoluto (MAE): Corresponde ao módulo do resíduo da predição para cada registro e é representado na Equação **(2)**. Será a média desses erros. Uma vantagem dessa métrica é que permite a mensuração do erro na mesma unidade medida, neste caso, dias.

DOI: 10.25286/repa.v9i1.2785

(2) $MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - p_j|$

- Erro médio logarítmico quadrático (MSLE): Mensura os resíduos em escala logarítmica e é útil quando há significativa variação na escala dos valores do alvo. É representado pela Equação **(3)**.

  (3) $MSLE = \frac{1}{n}\sum_{j=1}^{n}(\log(1+y_j) - \log(1+p_j))^2$

4 RESULTADOS

A Tabela 4 exibe os resultados encontrados após a parametrização dos modelos, treinamento e teste na base de dados utilizada. Em parênteses é exibido o desvio-padrão para as dez simulações executadas utilizando *k-fold* com *k* = 10 e com mesma semente de randomização dos registros. O modelo ingênuo serve como referência para os demais modelos em termos de melhoria de *performance*. Em geral, os modelos SVR e LGBM encontraram resultados interessantes, i.e., R^2 superior a 0.85. Entretanto, nota-se que os resultados encontrados pelo LGBM são ligeiramente superiores aos do SVR. Dessa forma, o modelo LGBM é capaz de reduzir o erro MAE do modelo ingênuo em aproximadamente quatro vezes e o MSLE em treze vezes. Enquanto os processos podem demorar de 1 a 9 anos, o LGBM é capaz de prever esse tempo com erro médio inferior a cinco meses, i.e., 143 dias.

Tabela 4 - Mensuração do desempenho dos modelos utilizando validação *k-fold* com *k = 10*

Modelo	R^2	MAE (dias)	MSLE
Ingênuo	-0.002 (0.002)	637 (18.934)	0.67 (0.071)
RL	0.652 (0.024)	358.512 (9.091)	0.261 (0.03)
SVR	0.864 (0.013)	170.289 (5.187)	0.07 (0.014)
LGBM	**0.905 (0.008)**	**142.839 (4.569)**	**0.05 (0.007)**

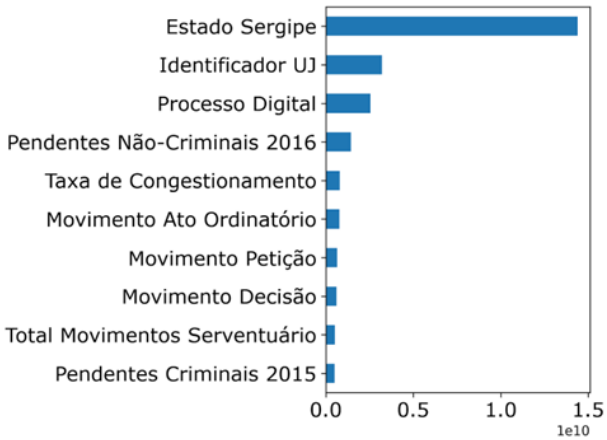
Fonte: Os autores.

4.1 IMPORTÂNCIA DAS FEATURES

O modelo LGBM permite a identificação das *features* mais relevantes durante a etapa de treinamento do modelo em termos de *gain* ou *split*. A importância da *feature* identificada por *gain* indica o ganho em termos de diminuição de erro causado pela inserção da *feature* nas árvores, i.e., *week learners,* de decisão do modelo. Esse erro pode ser quantificado por métricas como entropia ou índice gini. A importância *split* identifica o número de vezes que a *feature* foi utilizada como nó divisor em uma das árvores presentes no modelo. Foram utilizadas ambas as métricas para análise de relevância das *features*. A Figura 3 identifica as dez *features* mais relevantes para o modelo LGBM utilizando a importância *gain*.

Iremos analisar especialmente as quatro *features* mais relevantes para cada métrica. A *feature Estado Sergipe* é responsável por ganhos significativamente superiores às demais. Ela identifica se o processo judicial pertence a uma Unidade Judiciária do estado de Sergipe após aplicação de técnica de *one-hot-encoding*. Após investigação, notou-se que de fato o menor tempo médio processual pertence ao estado de Sergipe com uma média de 483 dias para finalização, enquanto que estados como Ceará apresentam uma média de 2.342 dias.

Figura 3 – As dez *features* mais relevantes em termos de *gain* para treinamento do modelo LGBM



Fonte: Os autores.

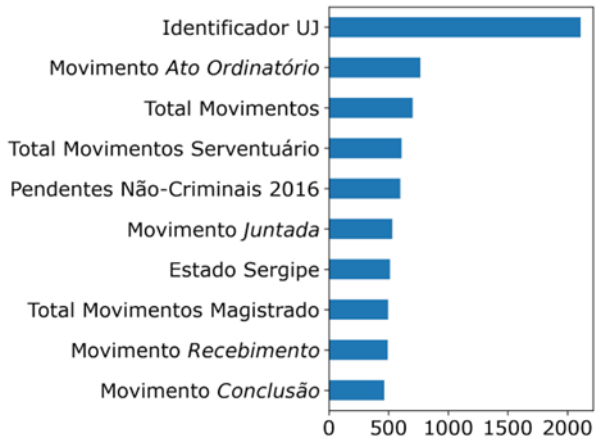
A *feature Identificador UJ* surpreende ao ser identificada como a segunda *feature* mais relevante, pois ela se refere somente ao código de identificação da Unidade Judiciária. Posterior investigação identificou que essa *feature* não possui numeração aleatória. Ao contrário, a numeração do *Identificador UJ* possui significativa correlação com características como UF e Classificação da UJ. Por exemplo, os identificadores das UJs analisadas variam entre 14.210 a 72.015. Entretanto, para as UJs do Ceará essa variação está apenas no intervalo entre 33.382 a 33.520. Similarmente, os identificadores de UJs do Rio Grande do Norte variam entre 43.717 a 43.792. As UJs de alguns estados como Paraíba apresentam maior variação

numérica, i.e., entre 14.232 a 44.872. Entretanto, se a *Classificação da Unidade* também for considerada é identificado um intervalo bem mais preciso, e.g., entre 14.232 a 14.308 para UJs da Paraíba classificadas como Zona Eleitoral. O modelo beneficiou-se do *Identificador UJ* pelo fato de essa *feature* não apresentar nulos ao passo que UF e Classificação continham ambas um percentual de 33% de nulos. A *feature Processo Digital* identifica se o processo foi executado por meios digitais ou físicos, i.e., com a utilização de "papéis". Foi observado que os 97% dos processos que foram executados por meio físico possuíram um tempo médio de 1.800 dias ao passo que nos 3% em que essa informação não foi identificada possuíram um tempo médio de 800 dias.

As *features* que iniciam por *Pendente* e a *Taxa de Congestionamento* identificam o número de processos pendentes e o nível de congestionamento da UJ responsável pelo processo judicial. Em especial, a feature *Pendentes Não-Criminais 2016* refere-se ao número de processos pendentes do tipo não-criminais no ano especificado. Identifica-se uma leve correlação de *Pearson* negativa, i.e., -0.24 entre essa *feature* e o tempo total processual. Isso poderia indicar que os processos analisados por essas UJs foram criminais e os não-criminais foram preteridos causando uma acumulação nos anos seguintes (já que para a maioria dos processos essa era uma variável *a posteriori*).

A Figura 4 identifica o número de vezes que a *feature* atuou como nó divisor em alguma das árvores de decisão do modelo LGBM. Nota-se que há seis *features* em comum com as da Figura 3 e quatro novas *features*.

Figura 4 – As dez *features* mais relevantes em termos de *split* para treinamento do modelo LGBM



Fonte: Os autores.

5 CONCLUSÕES

Neste trabalho foi realizada uma tarefa de mineração de dados e *feature engineering* para predição do tempo total processual. Os modelos SVR e LGBM obtiveram resultados interessantes na tarefa de predição do tempo total dos processos analisados em que o LGBM que obteve um R^2 *score* de 0.9 com desvio-padrão inferior a 0.01. Notou-se que *features* referentes a características da Unidade Judiciária responsável pelo processo como região (UF) e taxa de congestionamento destacaram-se entre as mais relevantes para a tarefa. Similarmente, *features* que identificaram o número e tipo de movimentações processuais ocorridas também foram relevantes. Os autores advogam que este tipo de análise pode gerar *insights* para melhorias processuais ao considerar simultaneamente características da UJ responsável pelo processo e as movimentações processuais ocorridas para o processo. Em trabalhos futuros, objetiva-se quantificar o ganho promovido pelas categorias de *features* bem como utilizar clusterização de processos como *features*.

REFERÊNCIAS

[1] LORIZIO, M.; GURRIERI, A. R. Efficiency of justice and economic systems. **Procedia Economics and Finance**, Elsevier, v. 17, p. 104–112, 2014.

[2] JOHNSEN, J. T. The european commission for the efficiency of justice (cepej) reforming european justice systems–'mission impossible?'. **International Journal for Court Administration**, v. 4, n. 3, 2012.

[3] SALUM, G. C. A duração dos processos no judiciário: aplicação dos princípios inerentes e sua eficácia no processo judicial. **Direito Processual Civil**, v. 145. 2016.

[4] VIEIRA, V. R. N. A morosidade do Judiciário, suas consequências para as partes e as formas de trazer celeridade aos processos no Brasil. In **Jusbrasil**, 2020. Disponível em < https://www.jusbrasil.com.br/artigos/a-morosidade-do-judiciario-suas-consequencias-para-as-partes-e-as-formas-de-trazer-celeridade-aos-processos-no-brasil/943683744> Acesso em 30 nov 2023.

DOI: 10.25286/repa.v9i1.2785

[5] GRUGINSKIE, L. A. d. S.; VACCARO, G. L. R. Lawsuit lead time prediction: Comparison of data mining techniques based on categorical response variable. **PloS one**, Public Library of Science San Francisco, CA USA, v. 13, n. 6, p. e0198122, 2018.

[6] MCCONNELL, D. J. et al. Case-level prediction of motion outcomes in civil litigation. In: Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law. [S.l.: s.n.]. p. 99–108. 2021

[7] LOEVINGER, L. Jurimetrics–the next step forward. **Minn. L. Rev.**, HeinOnline, v. 33, p. 455, 1948.

[8] ALETRAS, N. et al. Predicting judicial decisions of the european court of human rights: A natural language processing perspective. **PeerJ computer science**, PeerJ Inc., v. 2, p. e93, 2016.

[9] YASSINE, S.; ESGHIR, M.; IBRIHICH, O. Using artificial intelligence tools in the judicial domain and the evaluation of their impact on the prediction of judgments. **Procedia Computer Science**, Elsevier, v. 220, p. 1021–1026, 2023.

[10] AALST, W. M. Van der; SCHONENBERG, M. H.; SONG, M. Time prediction based on process mining. **Information systems**, Elsevier, v. 36, n. 2, p. 450–475, 2011.

[11] MAGGI, F. M. et al. Predictive monitoring of business processes. In: SPRINGER. **Advanced Information Systems Engineering: 26th International Conference, CAiSE 2014, Thessaloniki, Greece, June 16-20, 2014. Proceedings 26**. [S.l.]. p. 457–472. 2014.

[12] UNGER, A. J. et al. Process mining-enabled jurimetrics: analysis of a brazilian court's judicial performance in the business law processing. In: **Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law**. [S.l.: s.n.]. p. 240–244. 2021.

[13] D'CASTRO, R. J.; OLIVEIRA, A. L.; TERRA, A. H. Process mining discovery techniques in a low-structured process works? In: IEEE. **2018 7th Brazilian Conference on Intelligent Systems (BRACIS)**. [S.l.]. p. 200–205. 2018.

[14] OLIVEIRA, R. S. de; JR, A. S. R.; NASCIMENTO, E. G. S. Predicting the number of days in court cases using artificial intelligence. **PloS one**, Public Library of Science San Francisco, CA USA, v. 17, n. 5, p. e0269008, 2022.

[15] CAVNAR, W. B.; TRENKLE, J. M. et al. N-gram-based text categorization. In: LAS VEGAS, NV. **Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval**. [S.l.]. v. 161175, p. 14. 1994.

[16] KE, G. et al. Lightgbm: A highly efficient gradient boosting decision tree. **Advances in neural information processing systems**, v. 30, 2017.

[17] AWAD, M. et al. Support vector regression. **Efficient learning machines: Theories, concepts,and applications for engineers and system designers**, Springer, p. 67–80, 2015.