Embeddings Jurídico: Representações Orientadas à Linguagem Jurídica Brasileira

Fabrício A. do Carmo[1], **Ferdinando Serejo**[2], **Antonio F. L. Jacob Junior**[1], **Ewaldo E. C. Santana**[1], **Fábio M. F. Lobato**[1,3]

[1] Programa de Pós-Graduação em Engenharia de Computação e Sistemas (PECS)
Universidade Estadual do Maranhão - (UEMA)

[2]Tribunal de Justiça do Estado do Maranhão (TJMA)

[3]Instituto de Engenharia e Geociências
Universidade Federal do Oeste do Pará (UFOPA)

fabrycio30@gmail.com, fabio.lobato@ufopa.edu.br

Resumo. *O processamento automático de textos jurídicos dispostos em linguagem natural proporciona o desenvolvimento de diversas aplicações para o setor, como a classificação de processos por assunto, sumarização de documentos, tradução para linguagem cidadã etc. Nesse sentido, o judiciário brasileiro lançou o programa Justiça 4.0, buscando soluções que ofereçam celeridade nas atividades processuais. Convém pontuar que a linguagem técnica predomina nesse domínio de aplicação, o que adiciona desafios para modelagem dos dados, exigindo modelos especializados para o segmento. Frente ao exposto, esse trabalho tem como objetivo a construção de modelos embeddings orientados ao âmbito jurídico visando alimentar aplicações na área. Para isso, foram extraídos aproximadamente 500.000 documentos de instituições de justiça do Brasil das mais variadas esferas (civil, criminal, trabalhista etc). Os modelos foram avaliados por meio da classificação de petições iniciais e os resultados mostraram-se competitivos quando comparados a modelos generalistas da língua portuguesa. Tais resultados mostram que modelos treinados com documentos jurídicos compreendem melhor as especificidades da linguagem do segmento e têm o potencial de fomentar novas aplicações para o setor.*

1. Introdução

O uso de aplicações baseadas em Inteligência Artificial (IA) e *Big Data* vem apoiando a tomada de decisões em diversos segmentos da sociedade [Schaulet and Trez 2021, Garcia 2020, Hariri et al. 2019]. No âmbito jurídico, essas soluções podem guiar os profissionais tanto nas atividades administrativas quanto nos trâmites processuais, atuando principalmente sobre o grande volume de dados gerados no dia-a-dia da prestação jurisdicional [Pinto 2020, Parreiras et al. 2022]. No Brasil, onde o sistema judiciário conta com cerca de 77,3 milhões de processos em tramitação, segundo o último relatório "justiça em números"[1] do Conselho Nacional de Justiça (CNJ), já se entende que a celeridade processual passa necessariamente pela adoção de aplicações que utilizam recursos da IA.

[1]https://www.cnj.jus.br/pesquisas-judiciarias/justica-em-numeros/

O programa Justiça 4.0[2] é uma das iniciativas do CNJ que incentiva o desenvolvimento de soluções com tecnologias de IA, promovendo aquelas que visem a automatização e otimização dos serviços nos tribunais. Um exemplo dessas iniciativas é a plataforma SINAPSES[3], responsável tanto pelo estabelecimento de parâmetros (legais e técnicos) para o desenvolvimento e implantação de modelos de IA nos tribunais, como para o armazenamento e distribuição dos mesmos [Pereira and Rodrigues 2021].

Dentre as aplicações que utilizam os recursos da IA para análises de dados jurídicos, destacam-se os que consomem os recursos do Processamento de Linguagem Natural, conhecido por *Natural Language Processing* (NLP). NLP é uma área multidisciplinar que envolve a computação e a linguística e contempla estudos e desenvolvimento de métodos e procedimentos que permitam a compreensão e o processamento automático de textos dispostos na forma natural [Sousa and Del Fabro 2019, Hirschberg and Manning 2015]. Na seara jurídica, aplicações como classificação de documentos e processos [Polo et al. 2021, Bambroo and Awasthi 2021], do Reconhecimento de Entidades Nomeadas, ou *Named Entity Recognition* (NER) [Wang et al. 2020, Batista et al. 2021], já oferecem resultados práticos.

Um dos principais desafios no trato com documentos jurídicos está na compreensão das especificidades da linguagem utilizada pelo setor, composta por jargões e termos técnicos [Polo et al. 2021]. O tamanho dos textos jurídicos também é um aspecto significativo, geralmente composto de textos longos e dotado de formalismo [Bambroo and Awasthi 2021]. A exemplos, esses documentos são formados por decisões de julgamentos, contratos, normas legais, petições inicias etc [Zhong et al. 2020, Mota et al. 2020]. Dessa forma, treinar modelos de IA eficientes para esse domínio perpassa pela forma como os dados são tratados e estão representados.

As representações de dados textuais visam modelagens vetoriais representativas de um determinado texto. Elas são elementos fundamentais em NLP, dado a sua capacidade de transformar os documentos de entrada em vetores numéricos preservando informações originais e fornecendo entradas interpretáveis por modelos de *Machine Learning* (ML) e *Deep Learning* (DL) [Le-Khac et al. 2020]. Nesse sentido, os modelos *embeddings* são amplamente utilizados dado sua capacidade de adicionar informações semânticas no processo representacional [Chalkidis and Kampas 2019, Mikolov et al. 2013a]. Dentre os exemplos de modelos para representações para palavras e textos, destacam-se o *Word2vec* [Mikolov et al. 2013a], o *FastText* [Bojanowski et al. 2017] e, mais recentemente, o BERT [Devlin et al. 2019].

Os modelos baseados em *word embeddings* utilizam redes neurais rasas para o aprendizado dos vetores de representação. Tal procedimento permite o mapeamento de palavras e suas respectivas relações no *corpus* de treinamento, produzindo vetores que capturam informações semânticas e sintáticas advindas dessas relações contextuais [Mikolov et al. 2013a]. Dessa forma, por exemplo, as palavras "juiz" e "magistrado" estariam próximas em um espaço vetorial devido ao seu grau de similaridade.

Frente ao exposto, esta pesquisa busca fomentar o desenvolvimento de soluções baseadas no Processamento de Linguagem Natural na área jurídica brasileira, por meio

[2]https://www.cnj.jus.br/tecnologia-da-informacao-e-comunicacao/justica-4-0/

[3]https://www.cnj.jus.br/sistemas/plataforma-sinapses/

da construção de um *framework* experimental com diferentes modelos de representações *embeddings*, treinados a partir de documentos do setor. Os modelos treinados foram aplicados na classificação de petições inicias para avaliação de desempenho.

Destaca-se que a pesquisa reportada neste artigo faz parte do acordo de cooperação técnica entre a Universidade Estadual do Maranhão (UEMA) e o Tribunal de Justiça do Estado do Maranhão (TJMA), dessa forma, os modelos produzidos serão utilizados como insumos para o desenvolvimento de aplicações PLN no tribunal, gerando, dessa forma, impacto direto no sistema de justiça.

O restante do artigo encontra-se organizado como segue. Trabalhos relacionados são discutidos na Seção 2. A Seção 3 descreve o trabalho proposto. Na Seção 4 são apresentados os resultados e discussões. Por fim, na Seção 5 são apresentadas as conclusões e sugestões de trabalhos futuros.

2. Trabalhos relacionados

À luz da literatura do PLN, esta seção enfatiza trabalhos que apresentam técnicas e soluções com foco em representações *embeddings*, destacando aqueles direcionados a linguagem portuguesa e ao domínio jurídico. Inicialmente, são apresentadas as principais técnicas utilizadas. Na sequência, são mostrados os trabalhos com *embeddings* para a língua portuguesa, tanto para propósito geral e para a análise em domínios específicos. Por fim, são apresentados trabalhos direcionados ao âmbito jurídico.

2.1. Embeddings orientados a língua portuguesa

Um dos principais trabalhos com *embeddings* para língua portuguesa é o de [Hartmann et al. 2017]. Nele, os autores treinaram e disponibilizaram modelos FastText, GloVe, Wang2Vec e Word2Vec com diferentes dimensões, utilizando dados da língua portuguesa europeia e brasileira. Esses modelos foram analisados de forma intrínseca, por meio de análises sintáticas e semânticas, e extrínseca, utilizando-os em *Part-of-Speech* (PoS) *tagging* e análise de similaridades semânticas. Segundo os mesmos, a utilização de tarefas finais de NLP é preferível na avaliação dos modelos em detrimento a análises intrínsecas.

Em [Cunha et al. 2022] também é realizado treinamentos de modelos *embeddings* utilizando *corpus* da língua portuguesa, observando o impacto da parametrização dos modelos (e.g., dimensão do vetor), o tamanho do *corpus* de treinamento e do domínio. Também foram analisadas medidas para avaliação dos modelos treinados. Seus experimentos mostraram que as configurações paramétricas dos modelos têm influências significativas nos resultados. Também afirmam que avaliação por meio de analogias de palavras, utilizando *corpus* de teste, não é recomendada para modelos treinados em domínios mais segmentados, alinhando-se com as recomendações de [Hartmann et al. 2017].

Em uma análise mais segmentada, [Consoli et al. 2020] realizaram experimentos com *embeddings* treinadas com dados da área do petróleo e gás e fizeram comparações e combinações com modelos já treinados na língua portuguesa de modo geral, utilizando modelos disponibilizados por [Hartmann et al. 2017]. Os autores argumentaram que o setor contempla termos técnicos exclusivos, exigindo modelos mais orientados. Avaliações por meio do do NER, mostraram que combinações (*Stacking embeddings*) entre modelos

gerais e do domínio específico podem aumentar o desempenho da tarefa final, nesse caso alcançando *F1-score* de 84,63%.

Ainda no domínio de petróleo e gás, [Gomes et al. 2021] reforçam que a linguagem do setor possui características próprias e que palavras do português podem assumir significados completamente diferentes do comum, dificultando o aprendizado de algoritmos que consomem representações mais generalistas. Nesse trabalho, foram treinados modelos Word2vec e FastText em um *corpus* do domínio (contando com mais de 85 milhões de *tokens*). Os modelos foram submetidos a análises intrínsecas, analisando a relação entre pares de palavras, e extrínsecas, observando o desempenho no NER da área da Geociência. Os resultados mostraram que os modelos segmentados obtiveram melhores resultados quando comparados com os generalistas.

2.2. *Embeddings* Orientado ao segmento jurídico

Tal como no setor de petróleo e gás, a área jurídica compreende uma linguagem com características próprias na qual, por vezes, determinadas palavras possuem significados totalmente diferente da linguagem dita natural. Em [Smywiński-Pohl et al. 2019], são treinados modelos Word2vec e Glove e visando a criação de dicionário que forneça uma interface entre palavras técnicas da justiça polonesa e palavras que possam ser compreendidas por leigos. Os experimentos apontaram resultados superiores para o Word2vec do tipo CBOW. Também ressaltando essa peculiaridade no meio jurídico, [Polo et al. 2021] treinaram e disponibilizaram modelos de representações de palavras (Phraser, Word2Vec, Doc2Vec, FastText, e BERT), utilizando dados públicos da justiça brasileira. Realizaram experimentos com classificação de *status* (arquivado, ativo ou suspenso) de processos judiciais como demonstração de uso dos modelos treinados.

Em [Chalkidis and Kampas 2019], foram treinados e disponibilizados modelos *embeddings* a partir de um grande *corpus* de dados jurídicos disponíveis na língua inglesa. O *corpus* é formado por 123.066 peças jurídicas com aproximadamente 492.000.000 de *tokens* e envolve legislações do Reino Unido, União europeia, Canadá, Austrália, decisões da Suprema Corte Americana, além de documentos com legislações japonesas e da União europeia traduzidas para o inglês. As representações treinadas, nomeadas de law2vec[4], utilizaram o modelo word2vec com a arquitetura *Skip-gram*. Os autores afirmaram não adotar o *FastText* por ser tendencioso a informações sintáticas e dada a formalidade dos textos utilizados, com pouca incidência de erros, não haveria necessidade de adoção de um algoritmo que visa identificar/tratar palavras fora do vocabulário, buscando também contornar possíveis erros ortográficos.

No trabalho de [Wang et al. 2020], são utilizados modelos BERT como base para construção de arquiteturas híbridas para rotulagem de sequência (*sequence labelling*) orientadas à extração de entidades nomeadas. Foram utilizados dados jurídicos brasileiros no processo de avaliação dos modelos desenvolvidos. Ainda na seara da NER, [Batista et al. 2021] avaliaram o impacto de representações *embeddings* no processo de extração de entidades em petições inicias da justiça brasileira. Os resultados mostraram que a configuração com a *stacking* dos modelos *embeddings* de caracteres, de palavras e *pooled Flair* obteve melhores resultados.

[4]https://archive.org/details/Law2Vec

Também observando vetores *embedings* resultantes, [Dal Pont et al. 2020] avaliaram o impacto da especificidade e do tamanho do *corpus* de texto utilizado no treinamento dos vetores. Aplicados a dados jurídicos brasileiros em vários níveis de segmentação, os resultados mostraram que *corpus* menores capturam melhor as especificidades textos. Ou seja, para um ramo específico da justiça, nesse caso processos relacionados ao transporte aéreo), representações treinadas em *corpus* menores são preferíveis. Entretanto, de modo geral, quanto maior o *corpus* de treinamento, melhores são os resultados obtidos.

Os trabalhados supracitados destacam técnicas de representações textuais e aplicações envolvendo PLN que consomem tais recursos como entrada, também detalham medidas avaliativas para as soluções propostas. Os trabalhos segmentados realçam a importância de representações orientadas ao domínio do problema para obtenção de melhores resultados. Nesse sentido, o trabalho proposto foca na construção e treinamentos de modelos orientados ao âmbito jurídico que possa discriminar com maior eficácia as especificidades da linguagem desse domínio de aplicações.

3. Trabalho Proposto

O presente estudo teve como objetivo a construção e a divulgação de modelos *embeddings* orientados ao segmento jurídico brasileiro, com o intuito de fomentar aplicações NLP no setor. Tal como enfatizado em [Polo et al. 2021], a área jurídica compreende uma linguagem peculiar, requerendo representações que discriminem com maior eficiência o comportamento dos documentos jurídicos. Outro ponto que incentiva o desenvolvimento deste estudo é a falta de um volume significativo de dados disponíveis para o treinamento de modelos *embeddings* nesse domínio de aplicação. Destarte, este trabalho visa preencher essa lacuna, treinando modelos a partir de grande volume de dados públicos e privados da justiça brasileira. As subseções seguintes descrevem os dados, os procedimentos e as técnicas utilizadas para a construção e avaliação dos modelos adotados.

3.1. Dados Jurídicos

Para a criação do *corpus* jurídico de treinamento, tal como apresentado na Tabela 1, foram obtidos dados públicos contendo texto de acórdãos do Tribunal Superior do Trabalho (TST)[5]; do Supremo Tribunal Federal (STF), por meio do *Iudicium Textum Dataset* (ITD) disponibilizado por [Sousa and Del Fabro 2019]; do Superior Tribunal Militar (STM)[6]; do Tribunal Superior Eleitoral (TSE)[7]; do Tribunal de Contas da União (TCU)[8]; de processos recorrentes disponibilizados pelo Conselho Nacional de Justiça (CNJ) por meio do Banco Nacional de Dados de Demandas Repetitivas e Precedentes Obrigatórios (BNPR)[9]; e por meio de dados (e.g., decretos, acórdãos e súmulas) disponíveis na plataforma LexML[10], especializada na divulgação de informações jurídicas e legislativas; foram utilizados também dados internos, com textos de petições iniciais, do TJMA. Com exceção do ITD e do TJMA, os dados foram coletados diretamente dos portais, via *web crawlers*. A Tabela 1 apresenta características de cada *dataset*.

[5]https://www.tst.jus.br/jurisprudencia

[6]https://www.stm.jus.br/gestao-da-informacao/pagina-inicial-gest-inform/jurisprudencia

[7]http://www.tse.jus.br/jurisprudencia

[8]https://pesquisa.apps.tcu.gov.br/

[9]https://bnpr.cnj.jus.br/

[10]https://www.lexml.gov.br/

Tabela 1. Características dos conjuntos de dados jurídicos utlizados.

dataset	tipo do documento	amostras	*tokens*	*tokens* únicos
STF-ITD	Acórdãos STF	41.353	12.134.882	147.172
BNRP	Demandas recorrentes	3.255	179.439	15.126
TSE	Jurisprudência	84.754	8.284.523	109.070
STM	Jurisprudência	23.522	1.968.908	35.498
TCU	Jurisprudência	15.000	584.759	15.746
TST	Jurisprudência	247.084	488.794.965	856.537
STJ	Jurisprudência	1.772	122.338	8.743
LexML	Jurisprudência STF	72.280	2.502.145	54.811
TJMA	petições inicias	11.700	24.200.179	604.995
TOTAL	-	500.720	538.772.138	-

3.2. Modelos embeddings

1. **Word2vec**: É um algoritmo para geração de vetores de palavras, proposto por [Mikolov et al. 2013a], amplamente utilizado em PLN. O *Word2vec* tem a capacidade treinar vetores que consideram as relações entre as palavras, superando modelos mais simplificados como o *Bag-of-words* (BoW), onde a representação é focada em computar co-ocorrências dos termos [Qader et al. 2019]. A redução da dimensionalidade e o tratamento da esparsividade do vetor também são vantagens do modelo *word embeddings* sobre o BoW.
   O *Word2vec* utiliza duas estratégias de treinamento com redes neurais rasas: o *Continuous bag of words* (CBOW) e o modelo Skip-gram. No CBOW, o algoritmo tenta prever a palavra central (alvo), com base no contexto em que ela está inserida. Já no modelo Skip-gram a predição é feita de maneira oposta, as palavras do contexto são previstas com base na palavra central. As predições são realizadas utilizando janelas de contextos local, procedimento que seleciona k palavras vizinhas em torno do alvo.
   O modelo tem um custo computacional bem menor comparada a outras estratégias baseadas em redes neurais presentes na literatura e consegue obter vetores eficientes a partir de grandes conjuntos de dados [Mikolov et al. 2013a, Mikolov et al. 2013b].
2. **FastText**: É um modelo para representação de palavras proposto por [Bojanowski et al. 2017], tratado como uma extensão do modelo *Word2Vec*, que considera as palavras como um conjunto de *n*-gramas de caracteres. Dessa forma, o vetor resultante de uma determinada palavra é dado pela soma das subrepresentações de seus *n*-gramas. Essa abordagem de particionamento dos *tokens* possibilita a obtenção de representações para palavras não vistas no conjunto e treinamento (e.g., sufixos e prefixos). Além disso, palavras raras podem ter representações mais robustas do que aquelas obtidas pelos métodos *Word2Vec* [Polo et al. 2021].

3.3. Classificação de Petições Inicias

A petição inicial é um documento utilizado como primeiro passo para acessar ao Poder Judiciário, quando se está representado por Advogado(a), é nessa peça jurídica que está disposta a demanda requerida [Marinato et al. 2022]. Neste trabalho, utilizaram-se textos de petições iniciais fornecidas pelo TJMA para a avaliação das representações *embeddings* construídas. O objetivo é identificar (classificar) a qual Incidente de Resolução de

Demandas Repetitivas (IRDR), do referido tribunal, determinada petição inicial pertence. O IRDR é uma técnica utilizada pelos tribunais inferiores da justiça brasileira que visa fornecer a decisão para casos semelhantes, promovendo isonomia e segurança jurídica. O *dataset* utilizado já é anotado, ou seja, cada petição pertence a um determinado tipo de IRDR do TJMA. A Tabela 2 apresenta as características do conjunto de dados.

Tabela 2. Conjunto de dados de petições iniciais do TJMA.

Tipo de IRDR	Amostras	Incidência (%)
1	3581	65,18
2	27	0,49
3	502	9,14
4	54	0,98
5	1.068	19,44
6	4	0,07
7	6	0,11
8	252	4,59
TOTAL	5.494	100

3.4. Framework Experimental

Esta seção apresenta as etapas dos experimentos computacionais para o treinamento e avaliação dos modelos *embeddings* treinados a partir dos dados jurídicos. Essas etapas incluem: i) fase de tratamento dos textos de entrada; ii) fase de construção e treinamento dos modelos; iii) fase da análise dos modelos treinados.

Na fase de tratamento dos documentos jurídicos, foram aplicados filtros de pré-processamento de acordo com [Hartmann et al. 2017], realizando remoção de *stopwords*, espaços em branco, marcadores *HyperText Markup Language* (HTML) e quebra de linhas. Também foram configurados o reconhecimento de *tokens* de *e-mail* e *Uniform Resource Locator* (URL), além da transformação de caracteres para *lowercase*. Para essa fase, foram utilizados os recursos disponíveis na biblioteca Python Spacy[11],

Na fase de treinamento das representações, foram considerados diferentes configurações paramétricas para os modelos Word2Vec e FastText, visando ampla cobertura experimental. Para cada um dos modelos, foram analisados: i) tipo de arquitetura utilizada: CBOW e Skip-gram; ii) a dimensão do vetor de características: 300 e 600; e iii) épocas de treinamento: 10. Dessa forma, obtém-se uma representação para cada configuração experimental, como demostrado na Tabela 3. Foram utilizadas janelas de contexto de tamanho 5 e taxa de aprendizado de $0,03$. Para os demais parâmetros, foram utilizados os valores padrão da biblioteca.

Os parâmetros Dimensão e Número de Épocas de treinamento são adotados visando a uma análise de impacto na avaliação final, tal como discutido em [Cunha et al. 2022]. Observando se vetores maiores (com maior quantidade de características para determinada palavra) oferecerem melhores representações e se o impacto da quantidade de rodadas (épocas) de treinamento também é significativo, dado o aumento de custo computacional. Para análises comparativas, modelos já treinados na língua portuguesa também foram incorporados no *framework*, utilizados como *baseline* para os experimentos. Nesse caso, adotou-se os modelos treinados por [Hartmann et al. 2017] e

[11]https://spacy.io/

Tabela 3. Modelos utilizados nos experimentos computacionais.

id	modelo	dimensão	id	modelo-NILC	dimensão
C1	word2vec-Cbow	300	C1-NILC	word2vec-Cbow	300
C2	word2vec-Skip-gram	300	C2-NILC	word2vec-Skip-gram	300
C3	word2vec-Cbow	600	C3-NILC	word2vec-Cbow	600
C4	word2vec-Skip-gram	600	C4-NILC	word2vec-Skip-gram	600
C5	FastText-Cbow	300	C5-NILC	FastText-Cbow	300
C6	FastText-Skip-gram	300	C6-NILC	FastText-Skip-gram	300
C7	FastText-Cbow	600	C7-NILC	FastText-Cbow	600
C8	FastText-Skip-gram	600	C8-NILC	FastText-Skip-gram	600

disponibilizados no repositório do Núcleo Interinstitucional de Linguística Computacional (NILC)[12].

A fase de avaliação dos modelos consistiu na realização de experimentos aplicados em tarefas finais de NLP. Nesse caso, adotou-se a classificação de petições iniciais, tal como descrito na subseção anterior. Como métricas avaliativas, foram utilizadas a acurácia do classificador e, devido ao desbalanceamento das classes do conjunto de dados, a F1-macro. Para o aprendizado do modelo, utilizou-se validação cruzada do tipo *k-fold*, utilizando $k = 5$. No processo de classificação, foram adotados algoritmos de ML já bem conhecidos pela literatura: *Support Vector Machine (SVM)*, *Random Forest (RF)*, *K-Neighbors Neighbor (kNN)* e *Logistic Regression (LR)*, utilizando os recursos da biblioteca python Scikit-learn [13] para a implementação dos mesmos. Como hiperparâmetro do *k*-NN, adotou-se $k = 5$. Para o classificador SVM, foi utilizado a recurso *GridSearchCV* para seleção de parâmetros. Para os demais, foram adotados os paramêtros recomendados pela biblioteca.

4. Resultados e Análises

Esta seção apresenta e discute os resultados da avaliação dos modelos no processo de classificação de petições iniciais. Os resultados apontam que os vetores treinados foram superiores numericamente aos disponibilizados por [Hartmann et al. 2017].

A Tabela 4 mostra os resultados dos experimentos, destacando os melhores casos em negrito. Para a Acurácia (ACC), os modelos C2-NILC, C3-NILC e C4-NILC foram os que apresentaram os melhores resultados para a classificação, alcançando 76%, com o algoritmo *k*-NN. Já para F1-macro, os modelos treinados C6 e C7 foram os que ofereceram melhores resultados, com 42%, para o mesmo classificador. O algoritmo *k*-NN também foi o que obteve a melhor desempenho médio tanto para ACC quanto para F1-macro, com 74 e 40% para os modelos treinados e 75 e 39% para os do NILC, respectivamente. Os resultados médios da F1-macro também mostram um ganho de desempenho dos algoritmos no aprendizado de classes com menor incidência, quando utilizado os vetores treinados.

A Figura 1, apresenta a matriz de confusão com o aprendizado do classificador *k*-NN para o modelo C6, configuração com maior desempenho geral para F1-macro. Do ponto de vista de classificação, percebe-se dificuldades do algoritmo na predição para classes minoritárias. No entanto, ressalta-se que não foram aplicados mecanismos para

[12]http://www.nilc.icmc.usp.br/embeddings

[13]https://scikit-learn.org/stable

Tabela 4. Resultados dos experimentos para a classificação de IRDR.

id	*k*-NN		LR		RF		SVM	
	ACC	F1-macro	ACC	F1-macro	ACC	F1-macro	ACC	F1-macro
C1	0,75	0,40	0,72	**0,32**	0,66	0,12	0,66	0,15
C2	0,74	0,41	0,72	0,29	0,65	0,10	0,65	0,10
C3	0,75	0,40	0,72	0,31	**0,68**	0,16	**0,67**	**0,16**
C4	0,75	0,40	**0,73**	0,30	0,66	0,15	0,65	0,10
C5	0,73	0,36	0,71	0,28	0,67	0,15	0,65	0,10
C6	0,75	**0,42**	0,72	0,29	0,66	0,10	0,65	0,10
C7	0,73	**0,42**	0,71	0,30	0,67	0,17	0,66	0,13
C8	0,75	0,40	**0,73**	0,30	0,66	0,13	0,65	0,10
Média	0,74	**0,40**	0,72	**0,30**	**0,66**	**0,14**	**0,66**	**0,12**
C1-NILC	0,75	0,40	**0,73**	0,29	0,65	0,10	0,65	0,10
C2-NILC	**0,76**	0,41	**0,73**	0,29	0,65	0,13	0,65	0,10
C3-NILC	**0,76**	0,41	0,72	0,29	**0,68**	**0,19**	0,65	0,12
C4-NILC	**0,76**	0,41	**0,73**	0,31	0,66	0,15	0,65	0,10
C5-NILC	0,75	0,36	**0,73**	0,29	0,67	0,17	0,65	0,10
C6-NILC	0,74	0,38	**0,73**	0,29	0,65	0,10	0,65	0,10
C7-NILC	0,75	0,36	0,72	0,29	0,66	0,17	0,65	0,12
C8-NILC	0,74	0,39	0,72	0,28	0,66	0,11	0,65	0,10
Média	**0,75**	0,39	**0,73**	0,29	**0,66**	**0,14**	0,65	0,11

o balanceamento do conjunto de dados (Tabela 2) utilizados para classificação, por não ser o foco principal deste trabalho, deixando como ponto em aberto para experimentos futuros.

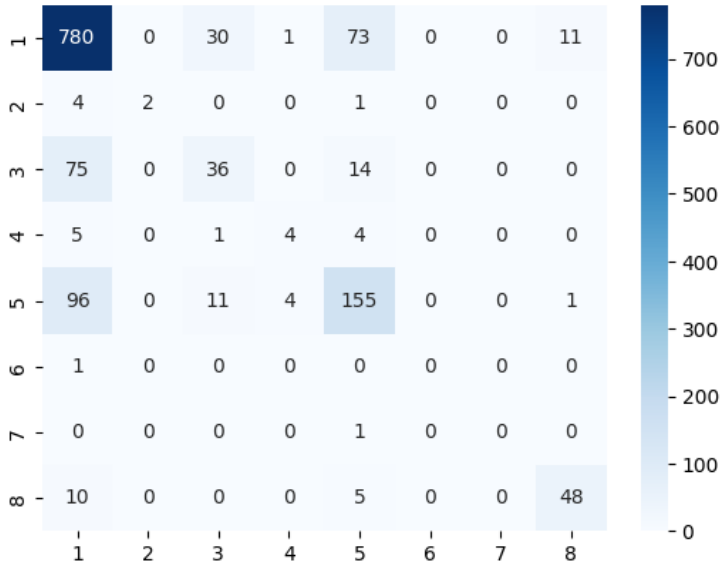

Figura 1. Matriz de Confusão do cenário com melhor desempenho (modelo C6 com algoritmo KNN, vide Tabela 4).

Os resultados obtidos mostram competitividade dos modelos treinados com os dados jurídicos em comparação aos disponibilizados por [Hartmann et al. 2017], treinados com grande volume de dados da língua portuguesa. A diferença no tamanho dos *corpus* indica que a utilização de documentos extraídos do domínio, tal como monstrado em [Gomes et al. 2021], pode fornecer representações que incorporam as especificidades da

área, utilizando quantidade de amostras relativamente menor, reduzindo também o custo computacional.

5. Conclusão

Em vista do Programa Justiça 4.0 do judiciário brasileiro, que busca soluções computacionais que ofereçam celeridade nas atividades processuais, o presente estudo apresentou a construção e avaliação de representações para palavras (*word embeddings*) orientadas à linguagem jurídica brasileira. As particularidades inerentes do domínio de aplicação adicionam complexidade para o aprendizado de algoritmos de *Machine Learning* e *Deep Learning*, principalmente para aqueles que recebem como entrada representações mais generalistas da língua portuguesa.

Como contribuição técnico-científica do estudo, além do *corpus* de mais de 500.000 documentos de instituições de justiça do Brasil das mais variadas esferas, disponibilizam-se os modelos de linguagem treinados com esse corpus e avaliação dos mesmos para a classificação de petições iniciais no repositório `https://github.com/fabiolobato/legal-embeddings-br`. Os resultados obtidos mostraram-se promissores quando comparados com modelos generalistas, indicando que modelos segmentados têm o potencial de melhorar sistemas inteligentes neste domínio. Destaca-se ainda que o presente estudo tem o potencial de fomentar o desenvolvimento de aplicações focadas no processamento de linguagem natural no domínio jurídico.

Ressaltamos que é uma pesquisa em andamento. Dessa forma, como próximos passos, pretendemos ampliar a cobertura experimental, incorporando: (i) outras técnicas de representação (e.g., Glove e o BERT); ii) novas estratégias de avaliação para além da classificação de dados, como agrupamento e mensuração de similaridade semântica; e iii) testes com outros modelos pré-treinados no domínio, como *baselines* comparativos.

Agradecimentos

Este trabalho foi apoiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq)-DT-308334/2020; pela Fundação Amazônia de Amparo a Estudos e Pesquisas (FAPESPA) PRONEM-FAPESPA/CNPq nº 045/2021; e pelo Acordo de Cooperação Técnica N° 02/2021 (Processo N° 38328/2020 -TJ/MA).

Referências

Bambroo, P. and Awasthi, A. (2021). Legaldb: Long distilbert for legal document classification. In *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, pages 1–4, Bhilai, India. 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT).

Batista, H., Nascimento, A., Melo, R., Miranda, P., Maldonado, I., and Filho, J. C. (2021). A comparative analysis of text embedding approach to extract named entities in portuguese legal documents. In *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*, pages 221–232, Porto Alegre, RS, Brasil. SBC.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146.

Chalkidis, I. and Kampas, D. (2019). Deep learning in law: early adaptation and legal word embeddings trained on large corpora. *Artificial Intelligence and Law*, 27(2).

Consoli, B., Santos, J., Gomes, D., Cordeiro, F., Vieira, R., and Moreira, V. (2020). Embeddings for named entity recognition in geoscience Portuguese literature. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4625–4630, Marseille, France. European Language Resources Association.

Cunha, L. F., Almeida, J. J. a., and Simões, A. (2022). Reasoning with Portuguese Word Embeddings. In Cordeiro, J. a., Pereira, M. J. a., Rodrigues, N. F., and Pais, S. a., editors, *11th Symposium on Languages, Applications and Technologies (SLATE 2022)*, volume 104 of *Open Access Series in Informatics (OASIcs)*, pages 17:1–17:14, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Dal Pont, T. R., Sabo, I. C., Hübner, J. F., and Rover, A. J. (2020). Impact of text specificity and size on word embeddings performance: An empirical evaluation in brazilian legal domain. In *Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I*, page 521–535.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Garcia, A. C. (2020). Ética e inteligencia artificial. *Computação Brasil*, 43:14–22.

Gomes, D. d. S. M., Cordeiro, F. C., Consoli, B. S., Santos, N. L., Moreira, V. P., Vieira, R., Moraes, S., and Evsukoff, A. G. (2021). Portuguese word embeddings for the oil and gas industry: development and evaluation. *Computers in Industry*, 124:103347.

Hariri, R. H., Fredericks, E. M., and Bowers, K. M. (2019). Uncertainty in big data analytics: survey, opportunities, and challenges. *Journal of Big Data*, 6(1):44.

Hartmann, N. S., Fonseca, E. R., Shulby, C. D., Treviso, M. V., Rodrigues, J. S., and Aluísio, S. M. (2017). Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Anais do XI Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, Porto Alegre, RS, Brasil. SBC.

Hirschberg, J. and Manning, C. D. (2015). Advances in natural language processing. *Science*, 349(6245):261–266.

Le-Khac, P. H., Healy, G., and Smeaton, A. F. (2020). Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934.

Marinato, M., Junior, A. J., Lobato, F., and Cortes, O. (2022). Classificação automática de petições iniciais usando classificadores combinados. In *Anais do XVI Brazilian e-Science Workshop*, pages 89–96, Porto Alegre, RS, Brasil. SBC.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Mota, C., Lima, A., Nascimento, A., Miranda, P., and de Mello, R. (2020). Classificação de páginas de petições iniciais utilizando redes neurais convolucionais multimodais. In *Anais do XVII Encontro Nacional de Inteligência Artificial e Computacional*, pages 318–329, Porto Alegre, RS, Brasil. SBC.

Parreiras, M., Vasconcellos, A., Mangeli, E., Yamamoto, E., Xexéo, G., Metello, I., Costa, L., Marques, P., and Souza, J. (2022). Inteligência artificial aplicada para o aumento da produtividade no atendimento de intimações. In *Anais do X Workshop de Computação Aplicada em Governo Eletrônico*, pages 180–191, Porto Alegre, RS, Brasil. SBC.

Pereira, J. C. M. and Rodrigues, M. V. J. (2021). A plataforma sinapses e a continuidade dos modelos de ia no judiciário. In *ANAIS do Encontro de Administração da Justiça - ENAJUS 2021*, Lisboa.

Pinto, H. A. (2020). A utilização da inteligência artificial no processo de tomada de decisões: por uma necessária accountability. *Revista de Informação Legislativa: RIL*.

Polo, F., Mendonça, G., Parreira, K., Gianvechio, L., Cordeiro, P., Ferreira, J., Lima, L., Maia, A., and Vicente, R. (2021). Legalnlp - natural language processing methods for the brazilian legal language. In *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*, pages 763–774, Porto Alegre, RS, Brasil. SBC.

Qader, W. A., Ameen, M. M., and Ahmed, B. I. (2019). An overview of bag of words;importance, implementation, applications, and challenges. In *2019 International Engineering Conference (IEC)*, pages 200–204.

Schaulet, E. and Trez, G. (2021). Big data em organizações de médio e grande porte do setor público brasileiro: Prontidão e situação atual, replicação do estudo holandês de klievink et al. (2017). In *Anais do IX Workshop de Computação Aplicada em Governo Eletrônico*, pages 13–24, Porto Alegre, RS, Brasil. SBC.

Smywiński-Pohl, A., Lasocki, K., Wróbel, K., and Strzała, M. (2019). Automatic construction of a polish legal dictionary with mappings to extra-legal terms established via word embeddings. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, ICAIL '19, page 234–238, New York, NY, USA. Association for Computing Machinery.

Sousa, A. W. and Del Fabro, M. D. (2019). Iudicium textum dataset uma base de textos jurıdicos para nlp. In *XXXIV Simpósio Brasileiro de Banco de Dados: Dataset Showcase Workshop, SBBD*, pages 1–11, Fortaleza, Brazil. SBBD.

Wang, Z., Wu, Y., Lei, P., and Peng, C. (2020). Named entity recognition method of brazilian legal text based on pre-training model. *Journal of Physics: Conference Series*, 1550(3):032149.

Zhong, H., Xiao, C., Tu, C., Zhang, T., Liu, Z., and Sun, M. (2020). How does NLP benefit legal system: A summary of legal artificial intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5218–5230, Online. Association for Computational Linguistics.